

ILM Library: Information Lifecycle Management Best Practices Guide

ILM basics

ILM building blocks

ILM strategies and solutions



Babette Haeusser
Alex Osuna
Christian Bosman
Dirk Jahn
Giulio John Tarella

Redbooks



International Technical Support Organization

ILM Library: Information Lifecycle Management Best Practices Guide

January 2007

Archived

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

Archived

First Edition (January 2007)

This edition applies to IBM storage products discussed at the time of this publication release.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	ix
Trademarks	x
Preface	xiii
The team that wrote this redbook	xiii
Become a published author	xvi
Comments welcome	xvi
Part 1. ILM basics	1
Chapter 1. Introducing ILM.	3
1.1 What ILM is	4
1.2 Why ILM is required	4
1.3 IT challenges and how ILM can help	8
1.4 ILM elements	10
1.4.1 Tiered storage management	11
1.4.2 Long-term data retention	13
1.4.3 Data lifecycle management	15
1.4.4 Policy-based archive management	17
1.5 Standards and organizations	18
1.6 IT Infrastructure Library and value of ILM	20
1.6.1 What is ITIL?	20
1.6.2 ITIL management processes	20
1.6.3 ITIL and ILM value	23
1.7 The technology layers of an ILM storage infrastructure	23
1.7.1 The storage hardware layer	24
1.7.2 The storage management layer	24
1.7.3 The information management middleware layer	25
Chapter 2. Planning for ILM	27
2.1 Business drivers: cost and efficiency	28
2.1.1 Challenges	28
2.1.2 The fluctuating value of data	30
2.1.3 Objectives	31
2.2 Focus areas	32
2.3 Taxonomy of legal requirements	36
2.3.1 Regulation examples	38
2.3.2 IBM ILM data retention strategy	39
2.4 Content management solutions	40
Part 2. ILM building blocks	41
Chapter 3. Information Management software	43
3.1 Content Management	44
3.1.1 Creation and capture of content	47
3.1.2 Management of content	47
3.1.3 Delivery of content	48
3.2 Choosing the right product for content repository	48
3.2.1 IBM DB2 Content Manager	48

3.2.2 IBM DB2 Content Manager OnDemand	53
3.3 Document management	54
3.3.1 IBM DB2 Document Manager	55
3.3.2 Lotus Domino Document Manager	56
3.4 IBM DB2 CommonStore	58
3.4.1 CommonStore for Exchange and CommonStore for Lotus Domino	58
3.4.2 CommonStore for SAP	59
3.5 IBM DB2 Records Manager	60
3.6 IBM Workplace Web Content Management	61
3.7 IBM Workplace Forms	62
3.8 Enterprise Search and Content Discovery	64
3.8.1 IBM WebSphere Information Integrator Content Edition	64
3.8.2 IBM WebSphere Information Integrator OmniFind Edition	67
3.8.3 IBM WebSphere Content Discovery Server	69
3.9 DB2 Content Manager VideoCharger	72
Chapter 4. IBM Tivoli Storage Manager and IBM System Storage Archive Manager ..	73
4.1 Tivoli Storage Manager concepts	74
4.1.1 Tivoli Storage Manager architectural overview	75
4.1.2 Tivoli Storage Manager storage management	82
4.1.3 Policy management	85
4.2 Hierarchical storage management	88
4.2.1 HSM in the Tivoli Storage Manager server	88
4.2.2 Space management for file systems	89
4.3 System Storage Archive Manager	92
4.3.1 Reasons for data retention	92
4.3.2 IBM System Storage Archive Manager	95
4.3.3 SSAM archive API options for data retention	98
4.3.4 Storage hardware options for Archive Manager	102
4.4 IBM System Storage N series SnapLock feature	103
4.4.1 SnapLock Compliance	103
4.4.2 SnapLock Enterprise	103
4.4.3 SSAM and IBM N series	104
4.4.4 IBM N series tiered storage	106
Chapter 5. Tiers of storage	111
5.1 Storage tiers	112
5.2 Enterprise disk systems	112
5.2.1 Storage consolidation	113
5.2.2 Performance	113
5.2.3 Data protection	115
5.2.4 Common set of functions	115
5.3 Midrange disk systems	116
5.4 IBM N series (Network Attached Storage)	121
5.4.1 Advantages of this storage solution	121
5.4.2 The IBM N series standard software features	122
5.4.3 Optional software	123
5.4.4 IBM System Storage N3700 Introduction	124
5.4.5 N5200 and N5500 Models A10 and A20	124
5.4.6 N5000 series gateway	124
5.5 Optical storage	127
5.6 Tape storage	128
5.6.1 LTO Ultrium tape drive	128

5.6.2	3592 J1A and TS1120 tape drives	130
5.6.3	Tape automation	134
5.7	Virtualization solutions	138
5.7.1	IBM TotalStorage SAN Volume Controller	139
5.7.2	IBM Virtualization Engine TS7510	140
	Chapter 6. IBM System Storage DR550	141
6.1	DR550 data retention solutions	142
6.1.1	IBM System Storage DR550	142
6.1.2	IBM System Storage DR550 Express	149
6.2	DR550 functions and capabilities	153
6.2.1	Flexible retention policies	153
6.2.2	Tiered storage solution and scalability	154
6.2.3	Data migration capabilities	154
6.2.4	Data encryption	154
6.2.5	Performance	154
6.3	ISV support list	155
6.3.1	IBM DB2 Content Manager	155
6.3.2	SSAM archive client	155
6.3.3	Other content management applications	156
	Part 3. Strategies and solutions	157
	Chapter 7. Assessing ILM	159
7.1	An ILM decision model	160
7.2	Best practices	165
7.2.1	Data rationalization	165
7.2.2	Storage virtualization	166
7.2.3	Tiered storage	168
7.2.4	Information management	169
7.2.5	Storage governance model	171
7.2.6	Archiving and information retention	173
7.3	The IBM approach with SMCD-ILM	176
	Chapter 8. IBM Tivoli Storage Manager best practices	179
8.1	Sizing the Tivoli Storage Manager environment	180
8.1.1	Determining business requirements	180
8.1.2	Sizing the Tivoli Storage Manager environment and selecting media	181
8.2	Business continuity and disaster recovery considerations	189
8.2.1	Protecting the server and the database	189
8.2.2	Protecting the Tivoli Storage Manager primary storage pools	192
8.2.3	Tivoli Storage Manager Disaster Recovery Manager (DRM)	194
8.2.4	Sample high availability and disaster recovery configurations	198
8.3	SSAM API essentials	204
8.3.1	Programming to the SSAM API	204
8.3.2	Application architectures	209
8.4	Using SSAM archive client for files	211
8.4.1	Archiving files with chronological retention	212
8.4.2	Archiving files for event based retention	213
8.4.3	SSAM and SnapLock best practices	214
	Chapter 9. Content Management and integrated Storage Management	217
9.1	Content and storage management product interactions	218
9.2	DB2 Content Manager, Tivoli Storage Manager, and SSAM	222

9.3	DB2 Content Manager OnDemand	229
9.4	DB2 CommonStore	236
9.5	Records and retention management	244
9.5.1	DB2 Records Manager integration into DB2 Content Manager	248
9.5.2	DB2 CM and Storage Management together with DB2 Records Manager	251
9.5.3	Use cases for the described configurations	252
	Chapter 10. File system archiving and retention	257
10.1	File systems	258
10.2	Archiving and retention	258
10.2.1	The archive client	258
10.2.2	Archiving and the SSAM and DR550	260
10.2.3	The TRIADE TriFSG DataGateway	261
10.3	Hierarchical storage management solutions	262
10.3.1	File systems and hierarchical storage management	262
10.3.2	IBM Tivoli Storage Manager for Space Management	265
10.3.3	IBM Tivoli Storage Manager for Space Management: UNIX	266
10.3.4	Tivoli Storage Manager for Space Management: Windows	268
10.3.5	Best practices in hierarchical storage management	270
10.4	IBM Tivoli CDP Continuous Data Protection	271
10.5	General Parallel Filesystem (GPFS)	272
10.5.1	GPFS architecture	272
10.5.2	GPFS Information Lifecycle Management	273
10.5.3	GPFS typical deployments	276
10.6	N series archiving and retention	278
10.6.1	N series SnapLock	278
10.6.2	N series LockVault	279
	Chapter 11. An introduction to GPFS	281
11.1	Overview	282
11.2	What is GPFS?	282
11.3	The file system	283
11.3.1	Application interfaces	284
11.3.2	Performance and scalability	284
11.3.3	Administration	285
11.3.4	Data availability	287
11.3.5	Information Lifecycle Management (ILM)	287
11.4	Cluster configurations	288
11.4.1	Shared disk	288
11.4.2	Network-based block IO	289
11.4.3	Sharing data between clusters	290
11.5	Summary	292
	Part 4. Appendixes	293
	Appendix A. DR550 services offerings	295
	QuickStart services for IBM System Storage DR550	296
	IBM RAID Conversion Services for IBM System Storage DR550	296
	Implementation Services for DR550	296
	Related publications	299
	IBM Redbooks	299
	Online resources	299
	How to get IBM Redbooks	299

Help from IBM	300
Index	301

Archived

Archived

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AFS®	FlashCopy®	Sametime®
AIX®	HACMP™	SLC™
AIX 5L™	IBM®	System i™
AS/400®	IMS™	System p5™
Domino®	Informix®	System x™
DB2®	iSeries™	System z9™
DB2 Universal Database™	Lotus®	System Storage™
DFSMSdss™	Lotus Notes®	Tivoli®
DFSMSHsm™	MVS™	TotalStorage®
DFSMSrmm™	Notes®	VideoCharger™
DS4000™	OmniFind™	Virtualization Engine™
DS6000™	OS/390®	WebSphere®
DS8000™	POWER™	Workplace™
e-business on demand®	POWER5™	Workplace Forms™
Enterprise Storage Server®	POWER5+™	Workplace Web Content Management™
ESCON®	pSeries®	xSeries®
eServer™	QuickPlace®	z/OS®
Eserver®	Redbooks™	zSeries®
Express Storage™	Redbooks (logo)  ™	z9™
FICON®	RS/6000®	

The following terms are trademarks of other companies:

SAP ArchiveLink, SAP NetWeaver, SAP R/3 Enterprise, mySAP.com, mySAP, SAP R/3, SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates.

Snapshot, SecureAdmin, SnapVault, SnapValidator, SnapRestore, SnapMover, SnapMirror, SnapManager, SnapDrive, FilerView, Data ONTAP, and the Network Appliance logo are trademarks or registered trademarks of Network Appliance, Inc. in the U.S. and other countries.

IT Infrastructure Library, IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

NetWeaver, mySAP.com, SAP, ArchiveLink, NetWeaver, mySAP.com, SAP R/3, SAP, R/3, mySAP, mySAP.com, xApps, xApp, SAP NetWeaver, and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world. All other product and service names mentioned are the trademarks of their respective companies.

Snapshot, SnapDrive, SecureAdmin, Data ONTAP, SnapVault, SnapRestore, SnapMover, SnapMirror, SnapManager, FilerView, The Network Appliance logo, the bolt design, Camera-to-Viewer, Center-to-Edge, ContentDirector, ContentFabric, NetApp Availability Assurance, NetApp ProTech Expert, NOW, NOW NetApp on the Web, RoboCache, RoboFiler, SecureAdmin, Serving Data by Design, Smart SAN, The evolution of storage, Virtual File Manager, and Web Filer are trademarks of Network Appliance, Inc. in the U.S. and other countries. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.

EJB, Java, J2EE, Solaris, StorageTek, Streamline, Sun, SLC, Ultra, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Excel, Microsoft, Outlook, Visual Basic, Visual C++, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Archived

Archived

Preface

This IBM® Redbook focuses on business requirements for information retention.

We provide practical recommendations for implementing a robust information management strategy. We also investigate the interactions of the various products and make recommendations for their use in different retention scenarios.

This book presents both a strategic and a practical approach. The strategy focuses on the value of ILM within an overall information management framework. The practical sections cover best practices for implementing and integrating ILM as a business process for long-term information retention.

The team that wrote this redbook

This IBM Redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Tucson Center.

Babette Haeusser is an IBM Certified IT Specialist at the International Technical Support Organization, San Jose Center. She writes extensively and teaches IBM classes worldwide on all areas of tape. Babette joined IBM in 1973 as an application programmer. In 1987, she became an MVS™ Systems Engineer and specialized in IBM Storage Hardware and Software, which she supported in various job roles since then. Before joining the ITSO in early 2005, Babette worked in the Advanced Technical Sales Support EMEA. She led a team of specialists for Enterprise Storage while focusing on Enterprise Tape, including tape libraries and Virtual Tape Servers.

Alex Osuna is a project leader at the International Technical Support Organization, Tucson. He writes extensively and also develops educational materials. Alex has over 28 years of experience in the IT industry with job roles in Maintenance, Field Engineering, Service Planning, Washington Systems Center, Product and Business planning, Advanced Technical Support, Systems Engineering, and his current role as Project Leader. Alex holds over 10 certifications with IBM, Microsoft®, and Red Hat.

Christian Bosman has a bachelor's degree in Electrical Engineering in Information Technology. He has been working in the IT industry for more than 11 years. Christian is an IBM IT Specialist providing storage field technical sales support in the Netherlands since 2001. He is specializing in removable media storage and data retention solutions. Christian advises customers, business partners, and IBM about storage (in general), tape, optical, and data retention solutions. Christian has the IBM Information Lifecycle Management and Information On Demand initiatives as a primary focus now.

Dirk Jahn is an IT Specialist working as a Content Management Presales Consultant for IBM Software Group in Germany. He has 10 years of experience in Content Management solutions in distributed environments. He holds a degree in Computer Science from the Institute of Technology in Goerlitz. His areas of expertise include IBM Content Management solutions, Records Management, and its integration into Tivoli® Storage Management and Storage networks.

John G. Tarella (John) is a Consulting IT Specialist who works for IBM Global Services in Italy. He has sixteen years of experience in storage and performance management on mainframe and distributed environments. He holds a degree in Seismic Structural Engineering from Politecnico di Milano, Italy. His areas of expertise include IBM Tivoli Storage Manager and SAN consulting, design, implementation services, and open systems storage, and storage performance monitoring and tuning. He is presently focusing on storage solutions for continuity, lifecycle management, and simplification. He has written extensively on z/OS® DFSMS, IBM Tivoli Storage Manager, and SANs.



The team: Chris, Babette, John, and Dirk



Alex Osuna

Thanks to the following people for their contributions to this project:

Charlotte Brooks, Bertrand Dufrasne, Wei-Dong Zhu, Emma Jacobs,
Yvonne Lyon, Leslie Parham, Deanna Polm, Sangam Racherla
International Technical Support Organization, San Jose Center

Chris Saul, Todd Neville, Alan Stuart, Errol Denger, Evan Salop, Timothy Waters, Kenneth
Nelson, Mark Kaplan, Robert Curran, Toby Marek, Tricia Jiang, Jarrett Potts, Cyrus Niltchian,
Imtiaz A Khan, Robert Constable, Chris Stakutis
IBM US

Andreas Luengen
IBM Germany

Francesco Conti
IBM Italy

Kai Nunnemann
Becom Informations systeme GmbH

Burghard Nuhn
TRIADE GmbH

Jenn Reese
Princeton Softech

Rob Gjersten
GPFS development

Reinhold Englebrecht
Robert Constable
Ken Nelson
Phillip Sanchez
Gerald Kozina
Larry Schroeder
Michael Heyl
Larry Heathcote
Imtiaz A Khan
Richard Hogg
Nick Kanellos
Joel Watermann
Brenda M. Brown
Martin Herbach
Andreas Kirchvogel
Henry Martens

Become a published author

Join us for a two-week to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts help increase product acceptance and client satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our IBM Redbooks™ to be as helpful as possible. Send us your comments about this or other IBM Redbooks in one of the following ways:

- ▶ Use the online **Contact us** review redbook form found at:

ibm.com/redbooks

- ▶ Send your comments in an e-mail to:

redbook@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

ILM basics

In this part of the book we discuss the following topics:

- ▶ The basic concepts of ILM
- ▶ What ILM is, and what storage management is
- ▶ Which components build an ILM solution, and how they interact with each other
- ▶ How to develop an ILM strategy

Archived

Introducing ILM

Information is essential to any business. Organizations have the challenge to efficiently manage information, throughout its lifecycle, related to its business value. The quantity of information and its value changes over time, and becomes increasingly costly and complex to store and manage.

This chapter discusses the importance of Information Lifecycle Management (ILM), its benefits, and introduces you to the elements of data lifecycle management. We introduce ILM and business drivers for adopting and building an ILM strategy. This chapter also provides an insight to:

- ▶ What an ILM is; for example, an important part of the IBM Information On Demand strategy
- ▶ How information and storage are managed, and the difference between ILM and data lifecycle management (DLM)
- ▶ What the business drivers for ILM are
- ▶ What the technology layers for an ILM solution are

1.1 What ILM is

Information Lifecycle Management (ILM) is a process for managing information through its lifecycle, from conception until disposal, in a manner that optimizes storage and access at the lowest cost.

ILM is not just hardware or software, it includes processes and policies to manage the information. It is designed upon the recognition that different types of information can have different values at different points in their lifecycle. Predicting storage requirements and controlling costs can be especially challenging as the business grows.

The overall objectives of managing information with Information Lifecycle Management are to help reduce the total cost of ownership (TCO) and help implement data retention and compliance policies. In order to effectively implement ILM, owners of the data are required to determine how information is created, how it ages, how it is modified, and if/when it can safely be deleted. ILM segments data according to value, which can help create an economical balance and sustainable strategy to align storage costs with businesses objectives and information value. The adoption of ILM technologies and processes, as shown in Figure 1-1, turns this strategy into a business reality.

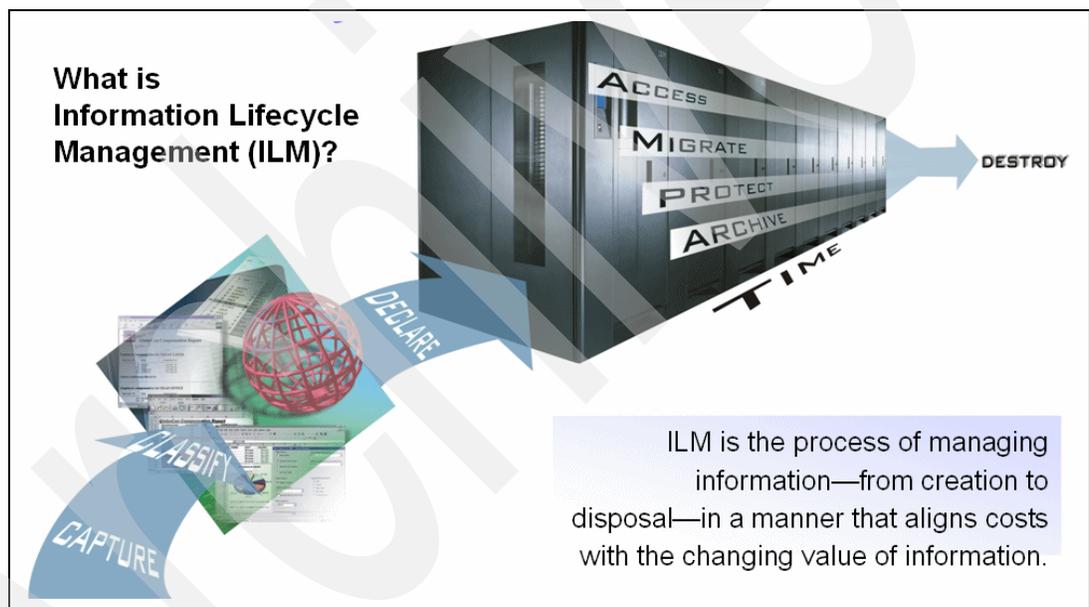


Figure 1-1 Information Lifecycle Management

1.2 Why ILM is required

In order to run your business efficiently, you require fast access to your stored data. But in today's business environment, you face increasing challenges: The explosion of the sheer volume of digital information, the increasing cost of storage management, tight regulatory requirements for data retention, and manual business and IT processes that are increasingly complex and error prone.

Although the total value of stored information has increased overall, historically, not all data is created equal, and the value of that data to business operations fluctuates over time. This is shown in Figure 1-2, and is commonly referred to as the *data lifecycle*. The existence of the data lifecycle means that all data cannot be treated the same.

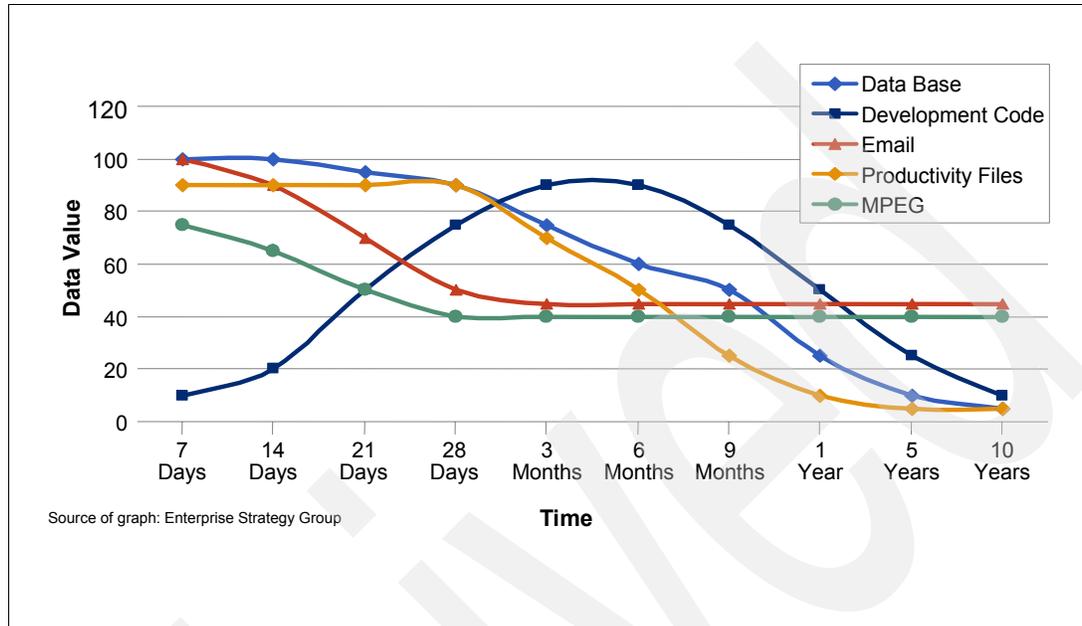


Figure 1-2 Data value changes over time

Figure 1-2 shows typical data values of different types of data, mapped over time. Most frequently, the value of data decreases over time, albeit at different rates of decline. However, infrequently accessed or inactive data can become suddenly valuable again as events occur, or as new business initiatives or projects are taken on. Historically, the requirement to retain information has resulted in a “buy more storage” mentality. However, this approach has only served to increase overall storage management costs and complexity, and has increased the demand for hard-to-find qualified personnel.

Executives today are tasked with reducing overall spending while supporting an ever increasing number of service and application demands. While support and management tasks increase, IT departments are being asked to justify their position by demonstrating business value to the enterprise. IT must also develop and enhance the infrastructure in order to support business initiatives while facing some or all of these data storage issues:

- ▶ Costs associated with e-mail management can reduce employee productivity in many companies.
- ▶ Backup and recovery windows continue to expand as data volumes grow unmanaged.
- ▶ Inactive data consumes valuable, high-performance disk storage space.
- ▶ Duplicate data copies are consuming additional storage space.
- ▶ As data continues to grow and management costs increase, budgets continue to be under pressure.

ILM entry points

Figure 1-3 represents the different starting points or entry points to create an ILM environment.

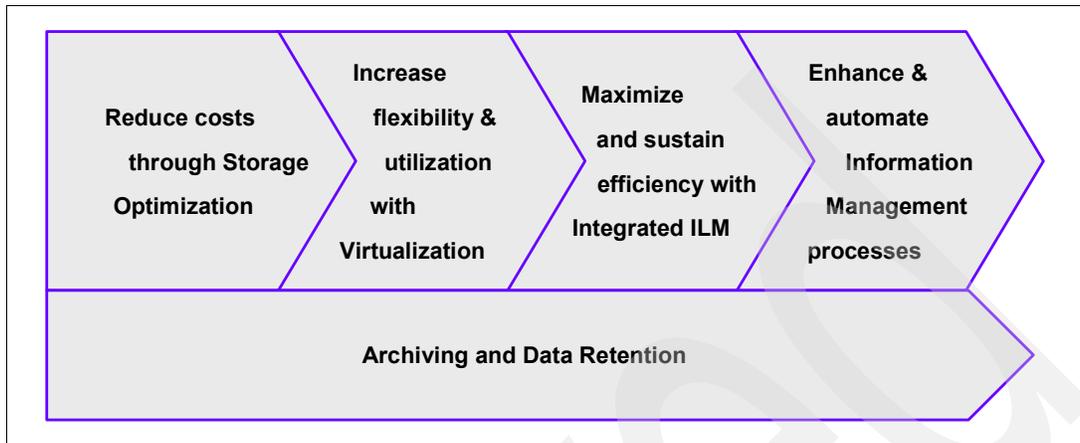


Figure 1-3 ILM infrastructure solution areas

This diagram introduces the components, or entry points, that installations have for building an ILM infrastructure. These represent different starting points. One benefit of this approach is that installations have the flexibility to start anywhere and begin getting results.

Installations do not have to be concerned with all of these aspects — just some of them, depending on what results they are seeking. Installations who are looking for the easiest ways to reduce cost tend to focus on the storage optimization entry point or the virtualization starting point. Installations who are looking for major efficiency plays are more likely to concentrate on the integrated ILM and/or starting points for the enhancement and automation of the information management process. Other installations, especially those who are concerned about compliance, are really looking at archiving and retention — although archiving and retention might also be considered by installations who are looking at the position of the “low hanging fruit”, where they can reduce their costs.

Multiple entry points provide a flexible approach to roll out a multi-year ILM strategy. Installations can zero in on more tactical IT projects to realize immediate returns while incrementally transitioning to an enterprise ILM strategy.

With an ILM solution, instead of blindly managing bits and bytes, installations can understand the importance of information to the business at each phase of its lifecycle, thus enabling them to align the business value of information with the most appropriate and cost effective IT infrastructure. Installations can also experience enhanced systems performance, both in the traditional sense and through faster applications, such as SAP® or e-mail, as well as from their infrastructure and storage systems.

Figure 1-4 shows the alignment of the various infrastructure solution entry points with the ILM best practices discussed in “ILM six best practices” on page 34.

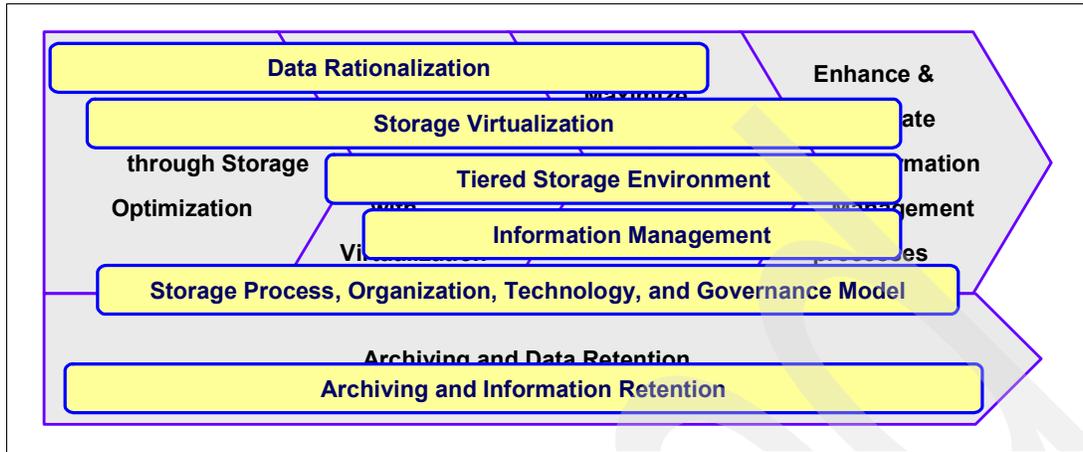


Figure 1-4 Best practices and ILM infrastructure solution areas

This diagram maps the starting points and shows where the best practices align with those starting points. It is a rough representation, not a perfect fit.

For example, in Figure 1-4, you see that data rationalization, as an initiative across that best practice, can fit across three of our starting points:

- ▶ Reduce costs
- ▶ Increase flexibility
- ▶ Maximize efficiency

Data rationalization initiatives are often a precursor to broader identity management projects in the areas of provisioning, authentication, and access control.

Virtualization is very relevant to all the starting points. Tiered storage tends to fit in to virtualization but also the integrated ILM and the enhanced automated processes. Virtualization is most commonly applied to servers, storage, and networks. It can also be applied to non-physical resources including applications, middleware, distributed systems, and even virtual resources themselves — for example, virtualizing a cluster of virtual servers. Although traditional resource virtualization continues to be a major component in IBM on demand strategy, for IBM to continue its thrust toward “Innovation through Virtualization”, more virtualization capability is required for creating virtual systems from multiple smaller systems, and for managing these systems across platform and vendor boundaries in a simplified, cohesive way.

Information management describes the programs, processes, architecture framework, standards and guidelines that the BT/CIO organization has designed to achieve effective management of data as a corporation-wide asset that meets the requirements of our external and internal customers. The primary objective of ILM is to support corporate-wide information and data management, including information warehouse management. Information management does definitely play a part in virtualization but really also extends into integrated ILM and the enhancement and automation of information management processes.

The storage process organization and technology and governance model plays all the way across these entry points. Storage optimization focuses on helping clients to improve the efficiency of the storage environment. The improved efficiency can include both an increased system utilization and/or personnel productivity. Particular techniques for increasing system utilization can include consolidation, virtualization, and automation. Personnel productivity techniques can include process, organization, technology, and governance.

Archiving and information retention can also be one of those best practices that installations implement in order to drive their particular results (Figure 1-5). Data archives are copies of active or inactive data from online storage, copied to offline storage. Archives are used to keep point-in-time copies of important data for regulatory or book-keeping requirements and to move inactive data from expensive online storage to less expensive offline storage.

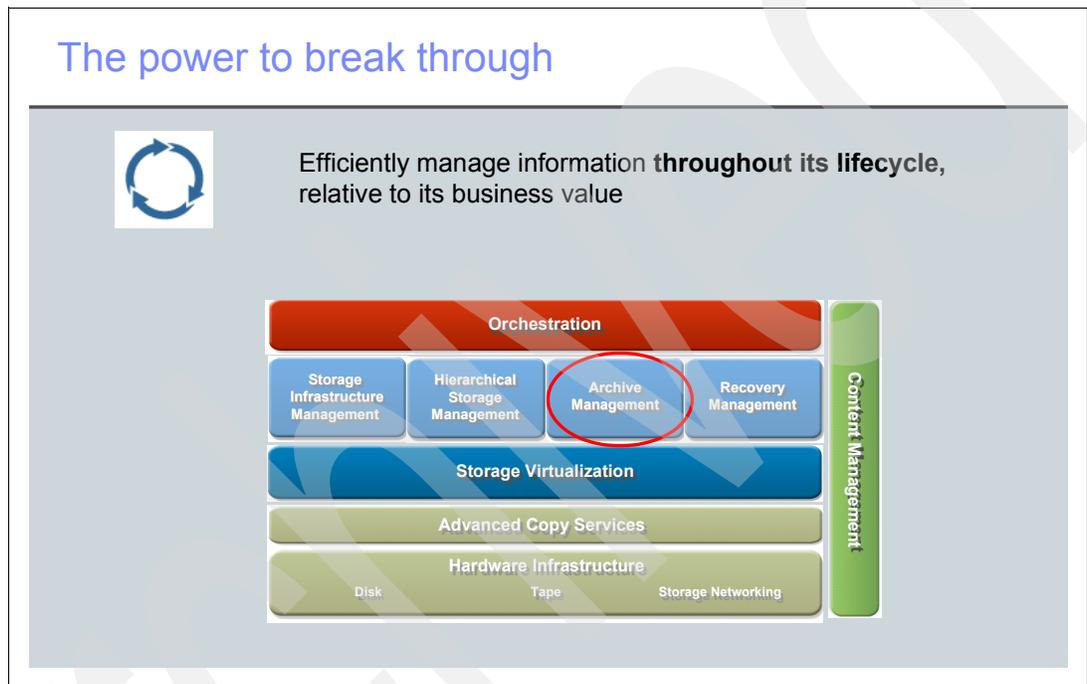


Figure 1-5 Archive management

1.3 IT challenges and how ILM can help

There are many challenges facing business today that make organizations think about managing their information more efficiently and effectively. Among these are some particular issues that might motivate you to develop an ILM strategy and solution:

- ▶ Information and data are growing faster than the storage budget.
- ▶ What data can I delete and when? What to keep and for how long?
- ▶ Disk dedicated to specific applications inhibits sharing.
- ▶ Duplicated copies of files and other data: Where are they, and how much space do they use?
- ▶ There is no mapping of the value of data to the value of the hardware on which it is stored.
- ▶ Longer time required to back up data, but the window keeps shrinking.
- ▶ Storage performance does not meet requirements.

- ▶ Low utilization of existing assets: For example, in open environments, storage utilization rates of around 30 percent are quite typical.
- ▶ Manual processes are causing potential business risk due to errors.
- ▶ Regulatory requirements dictate long-term retention for certain data.
- ▶ The business is unable to achieve backup/recovery/accessibility objectives for critical data.
- ▶ Inability to grow the support staff to keep up with the demand for storage management in an increasingly complex environment is a challenge.
- ▶ There are multiple backup and restore approaches and processes.
- ▶ Storage management requirements are not well defined.

In response to these challenges, it is necessary to define specific objectives to support and improve information management:

- ▶ Control demand for storage and create policies (Figure 1-6) for allocation.
- ▶ Reduce hardware, software, and storage personnel costs.
- ▶ Improve personnel efficiency, optimizing system, and productivity.
- ▶ Define and enforce policies to manage the lifecycle of data.
- ▶ Define and implement the appropriate storage strategy to address current and future business requirements.

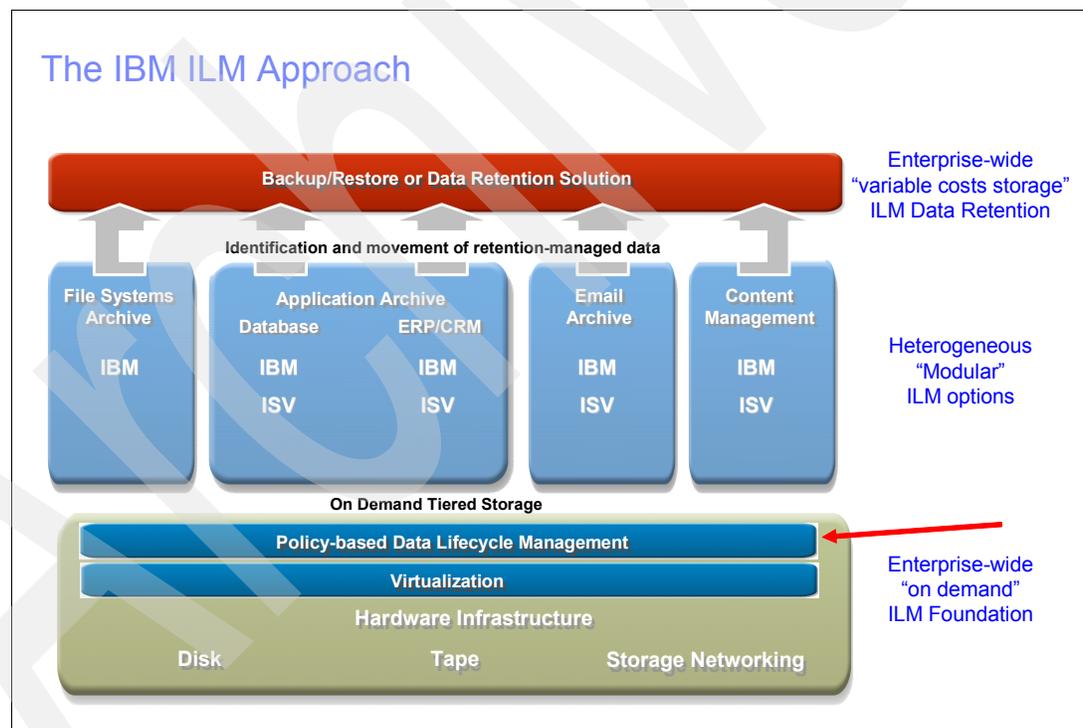


Figure 1-6 IBM ILM approach

The next section describes the major ILM solution components and how they can help you to overcome these challenges, and propose an ILM assessment for planning and design.

1.4 ILM elements

To manage the data lifecycle and make your business ready for On Demand, there are four main elements that can address your business to an ILM structured environment, as shown in Figure 1-7. They are:

- ▶ Tiered storage management
- ▶ Long-term data retention
- ▶ Data lifecycle management
- ▶ Policy-based archive management

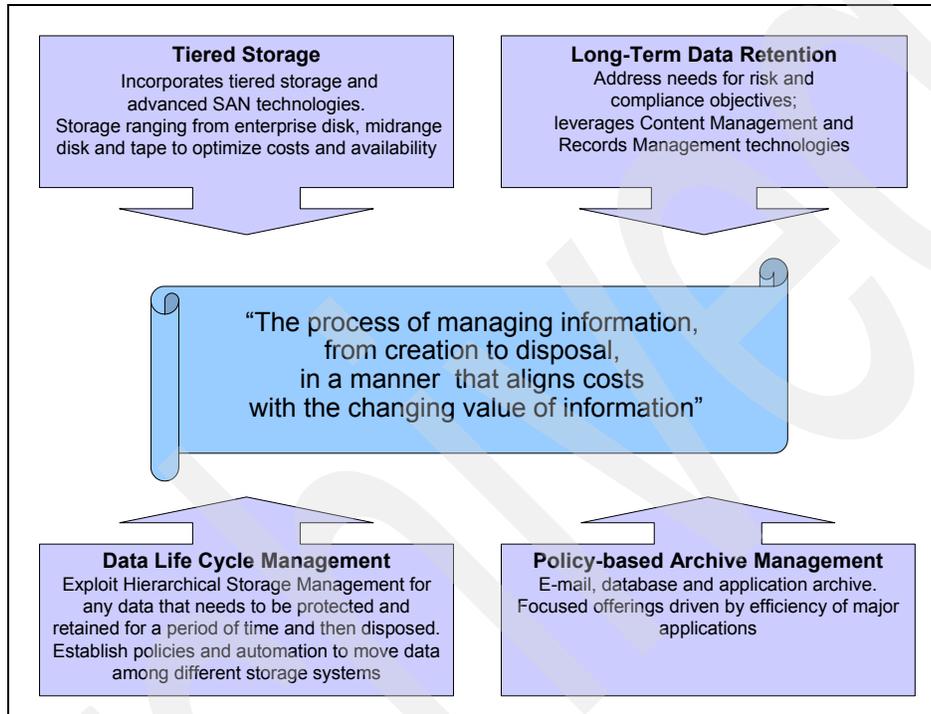


Figure 1-7 ILM elements

The next four sections describe each of these elements in detail:

- ▶ Tiered storage management
- ▶ Long-term data retention
- ▶ Data lifecycle management
- ▶ Policy-based archive management

1.4.1 Tiered storage management

Most organizations today seek a storage solution that can help them manage data more efficiently. They want to reduce the costs of storing large and growing amounts of data and files and maintain business continuity. Through tiered storage, you can reduce overall disk-storage costs, by providing benefits such as:

- ▶ Reducing overall disk-storage costs by allocating the most recent and most critical business data to higher performance disk storage, while moving older and less critical business data to lower cost disk storage.
- ▶ Speeding business processes by providing high-performance access to most recent and most frequently accessed data.
- ▶ Reducing administrative tasks and human errors. Older data can be moved to lower cost disk storage automatically and transparently.

Typical storage environment

Storage environments typically have multiple tiers of *data value*, such as application data that is required daily and archive data that is accessed infrequently. But typical storage configurations offer only a single tier of storage, as shown in Figure 1-8, which limits the ability to optimize cost and performance.

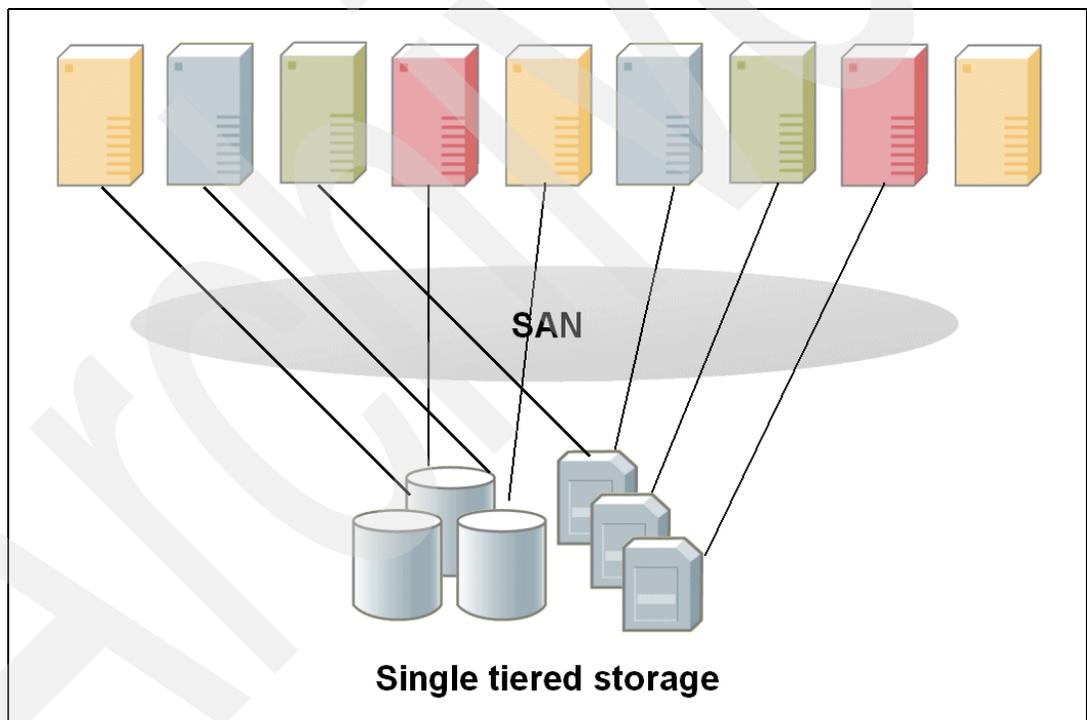


Figure 1-8 Traditional non-tiered storage environment

Multi-tiered storage environment

A tiered storage environment is the infrastructure required to align storage cost with the changing value of information. The tiers are related to data value. The most critical data is allocated to higher performance disk storage, while less critical business data is allocated to lower cost disk storage.

Each storage tier provides different performance matrix and disaster recovery capabilities. Creating classes and storage device groups is an important step to configure a tiered storage ILM environment. We provide details of this in later chapters of this book.

Figure 1-9 shows a multi-tiered storage environment.

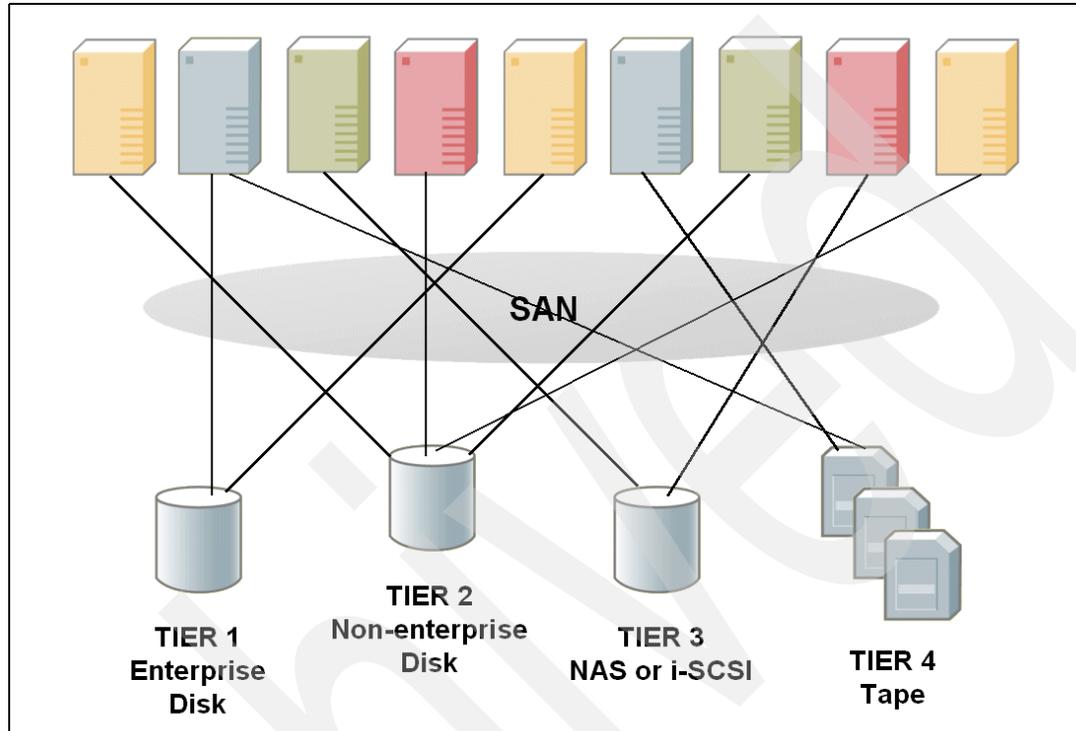


Figure 1-9 Multi-tiered storage environment

An IBM ILM solution in a tiered storage environment is designed to:

- ▶ Reduce the total cost of ownership of managing information. It can help optimize data costs and management, freeing expensive disk storage for the most valuable information.
- ▶ Segment data according to value. This can help create an economical balance and sustainable strategy to align storage costs with business objectives and information value.
- ▶ Help make decisions about moving, retaining, and deleting data, because ILM solutions are closely tied to applications.
- ▶ Manage information and determine how it must be managed based on content, rather than migrating data based on technical specifications. This approach can help result in a more responsive management, and offers you the ability to retain or delete information in accordance with business rules.
- ▶ Provide the framework for a comprehensive enterprise content management strategy.

Key products of IBM for tiered storage solutions and storage virtualization solutions are:

- ▶ IBM TotalStorage® SAN Volume Controller (SVC)
- ▶ IBM System Storage™ N series
- ▶ IBM TotalStorage DS family of disk storage, such as DS4x000, DS6000™, and DS8000™
- ▶ IBM TotalStorage tape drives, tape libraries, and virtual tape solutions

For details of these, see Chapter 5, “Tiers of storage” on page 111.

1.4.2 Long-term data retention

There is a rapidly growing class of data that is best described by the way in which it is managed rather than the arrangement of its bits. The most important attribute of this kind of data is its retention period, therefore it is called *retention managed data*, and it is typically kept in an archive or a repository. In the past it has been variously known as *archive data*, fixed content data, reference data, unstructured data, and other terms implying its read-only nature. It is often measured in terabytes and is kept for long periods of time, sometimes forever.

In addition to the sheer growth of data, the laws and regulations governing the storage and secure retention of business and client information are increasingly becoming part of the business landscape, making data retention a major challenge to any institution. An example of these is the Sarbanes-Oxley Act in the US, of 2002.

Businesses must comply with these laws and regulations. Regulated information can include e-mail, instant messages, business transactions, accounting records, contracts, or insurance claims processing, all of which can have different retention periods, for example, for 2 years, for 7 years, or retained forever. Moreover, some data must be kept just long enough and no longer. Indeed, content is an asset when it has to be kept. However, data kept past its mandated retention period could also become a liability. Furthermore, the retention period can change due to factors such as litigation. All these factors mandate tight coordination and the requirement for ILM.

Not only are there numerous state and governmental regulations that must be met for data storage, but there are also industry-specific and company-specific ones. And of course these regulations are constantly being updated and amended. Organizations have to develop a strategy to ensure that the correct information is kept for the correct period of time, and is readily accessible whenever regulators or auditors request it.

It is easy to envision the exponential growth in data storage that results from these regulations and the accompanying requirement for a means of managing this data. Overall, the management and control of retention managed data is a significant challenge for the IT industry when taking into account factors such as cost, latency, bandwidth, integration, security, and privacy.

Regulation examples

It is not within the scope of this book to enumerate and explain the regulations in existence today. For illustration purposes only, we list here some of the major regulations and accords in Table 1-1, summarizing their intent and applicability.

Table 1-1 Some regulations and accords affecting companies

Regulation	Intention	Applicability
SEC/NASD	Prevent securities fraud.	All financial institutions and companies regulated by the SEC
Sarbanes Oxley Act	Ensure accountability for public firms.	All public companies trading on a U.S. Exchange
HIPAA	Privacy and accountability for health care providers and insurers.	Health care providers and insurers, both human and veterinarian

Regulation	Intention	Applicability
Basel II aka The New Accord	Promote greater consistency in the way banks and banking regulators approach risk management across national borders.	Financial industry
21 CFR 11	Approval accountability.	FDA regulation of pharmaceutical and biotechnology companies

For example, in Table 1-2, we list some requirements found in SEC 17a-4 to which financial institutions and broker-dealers must comply. Information produced by these institutions, regarding solicitation and execution of trades and so on, is referred to as compliance data, a subset of retention-managed data.

Table 1-2 Some SEC/NASD requirements

Requirement	Met by
Capture all correspondence (unmodified) [17a-4(f)(3)(v)].	Capture incoming and outgoing e-mail before reaching users.
Store in non-rewritable, non-erasable format [17a-4(f)(2)(ii)(A)].	Write Once Read Many (WORM) storage of all e-mail, all documents.
Verify automatically recording integrity and accuracy [17a-4(f)(2)(ii)(B)].	Validated storage to magnetic, WORM.
Duplicate data and index storage [17a-4(f)(3)(iii)].	Mirrored or duplicate storage servers (copy pools).
Enforce retention periods on all stored data and indexes [17a-4(f)(3)(iv)(c)].	Structured records management.
Search/retrieve all stored data and indexes [17a-4(f)(2)(ii)(D)].	High-performance search retrieval.

IBM ILM data retention strategy

Regulations and other business imperatives, as we just briefly discussed, stress the requirement for an Information Lifecycle Management process and tools to be in place. The unique experience of IBM with the broad range of ILM technologies, and its broad portfolio of offerings and solutions, can help businesses address this particular requirement and provide them with the best solutions to manage their information throughout its lifecycle. IBM provides a comprehensive and open set of solutions to help.

IBM has products that provide content management, data retention management, and sophisticated storage management, along with the storage systems to house the data. To specifically help companies with their risk and compliance efforts, the IBM Risk and Compliance framework is another tool designed to illustrate the infrastructure capabilities required to help address the myriad of compliance requirements. Using the framework, organizations can standardize the use of common technologies to design and deploy a compliance architecture that might help them deal more effectively with compliance initiatives.

For more details about the IBM Risk and Compliance framework, visit:

<http://www-306.ibm.com/software/info/openenvironment/rcf/>

Here are some key products of IBM for data retention and compliance solutions:

- ▶ IBM Tivoli® Storage Manager, including IBM System Storage Archive Manager
- ▶ IBM DB2® Content Manager Family, which includes DB2 Content Manager, Content Manager OnDemand, CommonStore for Exchange Server, CommonStore for Lotus® Domino®, and CommonStore for SAP
- ▶ IBM System Storage N series
- ▶ IBM DB2 Records Manager
- ▶ IBM TotalStorage DS4000™ with SATA disks
- ▶ IBM System Storage DR550
- ▶ IBM TotalStorage Tape (including WORM) products

For details on these products, see Chapter 4, “IBM Tivoli Storage Manager and IBM System Storage Archive Manager” on page 73.

Important: The IBM offerings are intended to help clients address the numerous and complex issues relating to data retention in regulated and non-regulated business environments. Nevertheless, each client’s situation is unique, and laws, regulations, and business considerations impacting data retention policies and practices are constantly evolving. Clients remain responsible for ensuring that their information technology systems and data retention practices comply with applicable laws and regulations, and IBM encourages clients to seek appropriate legal counsel to ensure their compliance with those requirements. IBM does not provide legal advice or represent or warrant that its services or products are going to ensure that the client is in compliance with any law.

1.4.3 Data lifecycle management

At its core, the process of ILM moves data up and down a path of tiered storage resources, including high-performance, high-capacity disk arrays, lower-cost disk arrays such as serial ATA (SATA), tape libraries, and permanent archival media where appropriate. However, ILM involves more than just data movement, it encompasses scheduled deletion and regulatory compliance as well. Because decisions about moving, retaining, and deleting data are closely tied to application use of data, ILM solutions are usually closely tied to applications.

ILM has the potential to provide the framework for a comprehensive information-management strategy, and helps ensure that information is stored on the most cost-effective media. This helps enable administrators to make use of tiered and virtual storage, as well as process automation. By migrating unused data off of more costly, high-performance disks, ILM is designed to help:

- ▶ Reduce costs to manage and retain data.
- ▶ Improve application performance.
- ▶ Reduce backup windows and ease system upgrades.
- ▶ Streamline™ data management.
- ▶ Allow the enterprise to respond to demand, in real-time.
- ▶ Support a sustainable storage management strategy.
- ▶ Scale as the business grows.

ILM is designed to recognize that different types of information can have different values at different points in their lifecycle. As shown in Figure 1-10, data can be allocated to a specific storage level aligned to its cost, with policies defining when and where data is to be moved.

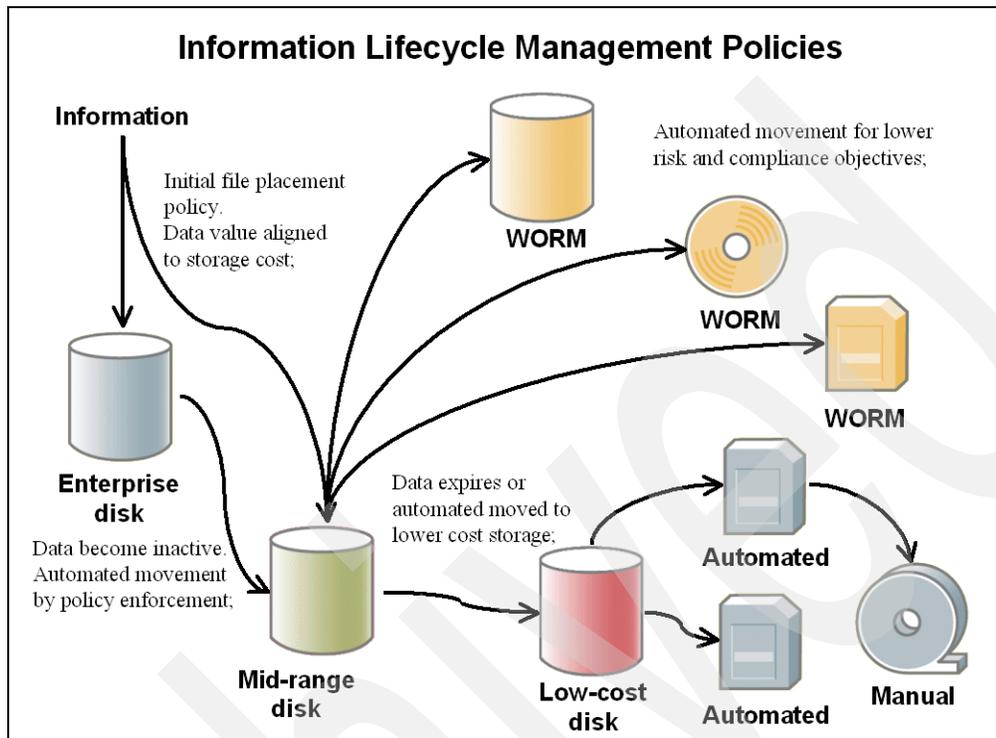


Figure 1-10 ILM policies

But, sometimes, the value of a piece of information might change, and data that was previously inactive and was migrated to a lower-cost storage now could be required and must be processed on a high-performance disk. A data lifecycle management policy can be defined to move the information back to enterprise storage, making the storage cost aligned to data value, as illustrated in Figure 1-11.

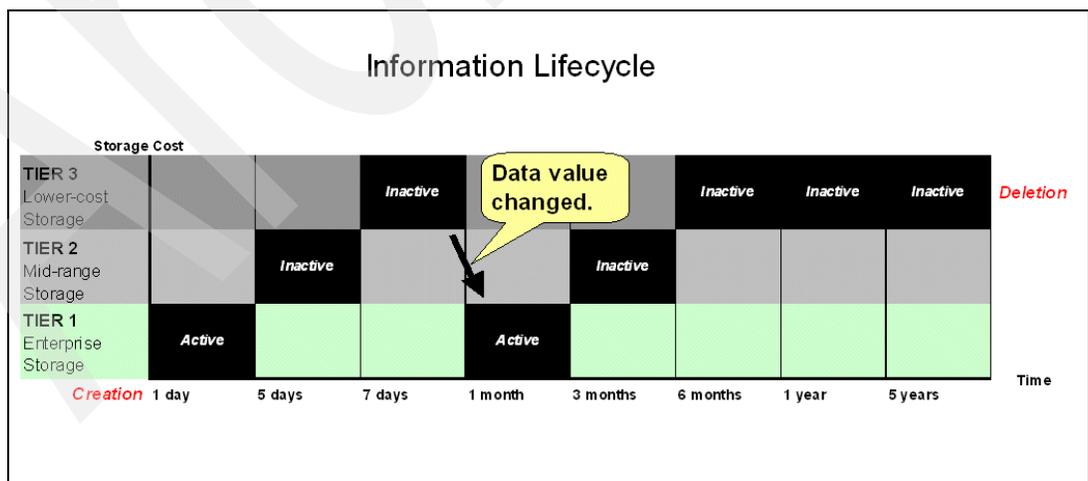


Figure 1-11 Information value changes

Key products of IBM for lifecycle management are:

- ▶ IBM TotalStorage Productivity Center
- ▶ IBM TotalStorage SAN Volume Controller (SVC)
- ▶ IBM Tivoli Storage Manager, including IBM System Storage Archive Manager
- ▶ IBM Tivoli Storage Manager for Space Management

For details of these products, see Chapter 5, “Tiers of storage” on page 111.

1.4.4 Policy-based archive management

As businesses of all sizes migrate to e-business solutions and a new way of doing business, they already have mountains of data and content that have been captured, stored, and distributed across the enterprise. This wealth of information provides a unique opportunity. By incorporating these assets into e-business solutions, and at the same time delivering newly generated information media to their employees and clients, a business can reduce costs and information redundancy and leverage the potential profit-making aspects of their information assets.

Growth of information in corporate databases such as Enterprise Resource Planning (ERP) systems and e-mail systems can make organizations think about moving unused data off the high-cost disks. They must now:

- ▶ Identify database data that is no longer being regularly accessed and move it to an archive where it remains available.
- ▶ Define and manage what to archive, when to archive, and how to archive from the mail system or database system to the back-end archive management system.

Database archive solutions can help improve performance for online databases, reduce backup times, and improve application upgrade times.

E-mail archiving solutions are designed to reduce the size of corporate e-mail systems by moving e-mail attachments and/or messages to an archive from which they can easily be recovered if required. This action helps reduce the requirement for end-user management of e-mail, improves the performance of e-mail systems, and supports the retention and deletion of e-mail.

The way to do this is to migrate and store all information assets into an e-business enabled content manager. ERP databases and e-mail solutions generate large volumes of information and data objects that can be stored in content management archives. An archive solution allows you to free system resources, while maintaining access to the stored objects for later reference. Allowing it to manage and migrate data objects gives a solution the ability to have ready access to newly created information that carries a higher value, while at the same time still being able to retrieve data that has been archived on less expensive media, as shown in Figure 1-12.

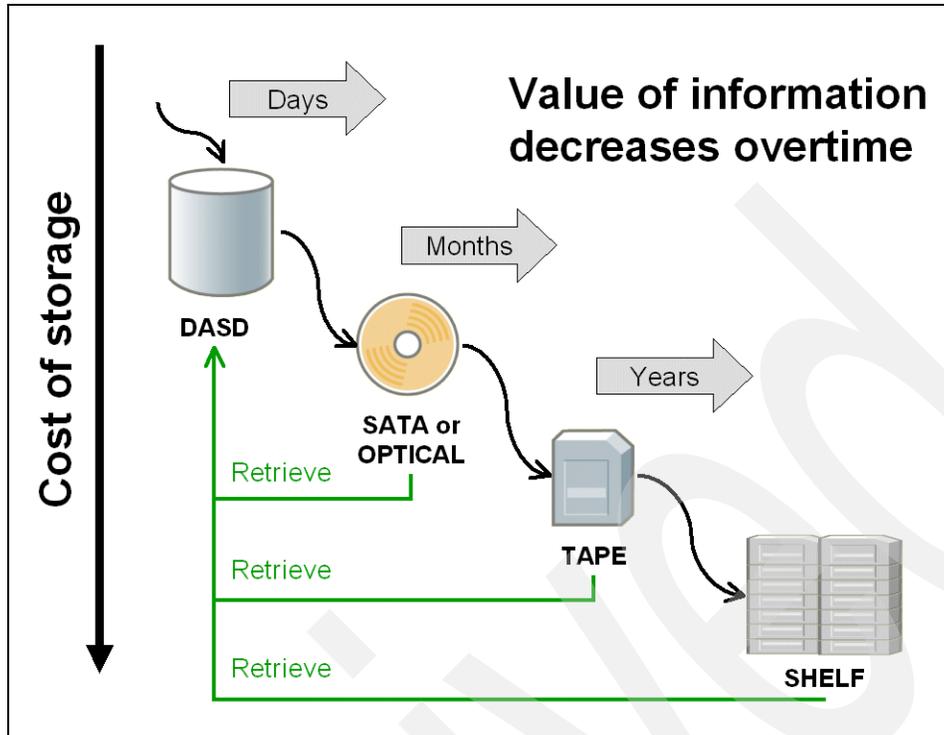


Figure 1-12 Value of information and archive/retrieve management

Key products of IBM for archive management are:

- ▶ IBM Tivoli Storage Manager, including IBM System Storage Archive Manager
- ▶ IBM DB2 Content Manager family of products
- ▶ IBM DB2 CommonStore family of products

For details about these products, see Chapter 5, “Tiers of storage” on page 111.

1.5 Standards and organizations

The success and adoption of any new technology, and any improvement to existing technology, is greatly influenced by standards. Standards are the basis for the interoperability of hardware and software from different, and often rival, vendors. Although standards bodies and organizations such as the Internet Engineering Task Force (IETF), American National Standards Institute (ANSI), and International Organization for Standardization (ISO) publish these formal standards, other organizations and industry associations, such as the Storage Networking Industry Association (SNIA), play a significant role in defining the standards and market development and direction.

Storage Networking Industry Association

The Storage Networking Industry Association is an international computer system industry forum of developers, integrators, and IT professionals who evolve and promote storage networking technology and solutions. SNIA was formed to ensure that storage networks become efficient, complete, and trusted solutions across the IT community. IBM is one of the founding members of this organization. SNIA is uniquely committed to networking solutions into a broader market. SNIA is using its Storage Management Initiative (SMI) and its Storage Management Initiative Specification (SMI-S) to create and promote adoption of a highly functional interoperable management interface for multivendor storage networking products.

SMI-S makes multivendor storage networks simpler to implement and easier to manage. IBM has led the industry in not only supporting the SMI-S initiative, but also using it across its hardware and software product lines. The specification covers fundamental operations of communications between management console clients and devices, auto-discovery, access, security, the ability to provision volumes and disk resources, LUN mapping and masking, and other management operations.

Data Management Forum

SNIA has formed the Data Management Forum (DMF) to focus on defining, implementing, qualifying, and teaching improved methods for the protection, retention, and lifecycle management of data.

Vision for ILM by SNIA and DMF

The Data Management Forum defines ILM as a new management practice for the datacenter. ILM is not a specific product, nor is it just about storage and data movement to low-cost disk. It is a standards-based approach to automating datacenter operations by using business requirements, business processes, and the value of information to set policies and service level objectives for how the supporting storage, compute, and network infrastructure operate.

The key question that flows from this *vision* of ILM is *How do we get there?*, because these capabilities do not fully exist today. This is the work of SNIA and the Data Management Forum. To unify the industry towards a common goal, to develop the relevant standards, to facilitate interoperability, and to conduct market education around ILM. Figure 1-13 illustrates the SNIA vision for ILM.

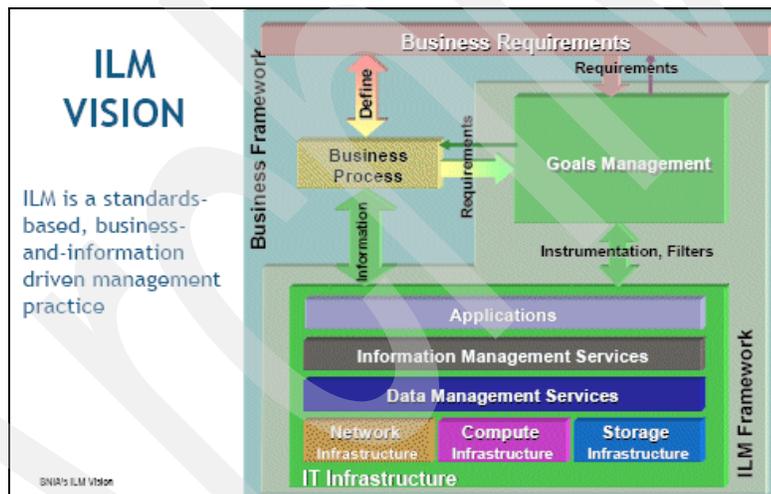


Figure 1-13 SNIA vision for ILM

For additional information about the various activities of SNIA and DMF, see this Web site:

<http://www.snia.org>

1.6 IT Infrastructure Library and value of ILM

The intent of this section is to introduce you to the IT Infrastructure Library® (ITIL®¹) and the value of ILM within the ITIL methodology. It begins by defining ITIL and its Service Support processes.

1.6.1 What is ITIL?

ITIL is a process-based methodology used by IT departments to verify that they can deliver IT services to end users in a controlled and disciplined way. It incorporates a set of best practices that are applicable to all IT organizations, no matter what size or what technology is used. ITIL is used to create and deliver service management processes. These tasks are made easier by the use of service and system management tools.

Over recent decades, multiple IT process models have been developed. ITIL is the only one that is not proprietary:

- ▶ Late 1970s: Information Systems Management Architecture (ISMA) (IBM)
- ▶ Late 1980s: IT Infrastructure Library V1 (ITIL) (CCTA - now OGC)
- ▶ 1995: IT Process Model (ITPM) (IBM)
- ▶ 2000: Enterprise Operational Process Framework (IBM)
- ▶ 2000: IT Service Management Reference Model (HP)
- ▶ 2000–2001: Microsoft Operations Framework (MOF) (Microsoft)
- ▶ 2001–2002: IT Infrastructure Library V2 (ITIL) (OGC)

Note: OGC is the UK Government's Office of Government Commerce. CCTA is the Central Computer and Telecommunications Agency.

ITIL has a library of books describing best practices for IT services management that describe goals, activities, and inputs and outputs of processes. It is a set of best practices. ITIL has a worldwide approach to IT management and its methodology sets that specific procedures can vary from organization to organization. ITIL is not tied to any particular vendor, and IBM has been involved with ITIL since its inception in 1988.

1.6.2 ITIL management processes

The ITIL approach to creating and managing *service management* processes is widely recognized around the world and the adoption of its principles is clearly growing, as evidenced by new groups appearing in more countries every year.

The *service management* disciplines are grouped into the two areas of Service Support and Service Delivery. There are now eleven basic processes used in the areas of Service Support and Service Delivery, as shown in Figure 1-14. Because it can take a long time to implement these disciplines, it is not uncommon to find only some of the processes in use initially.

¹ ITIL is a registered trademark of the OGC.

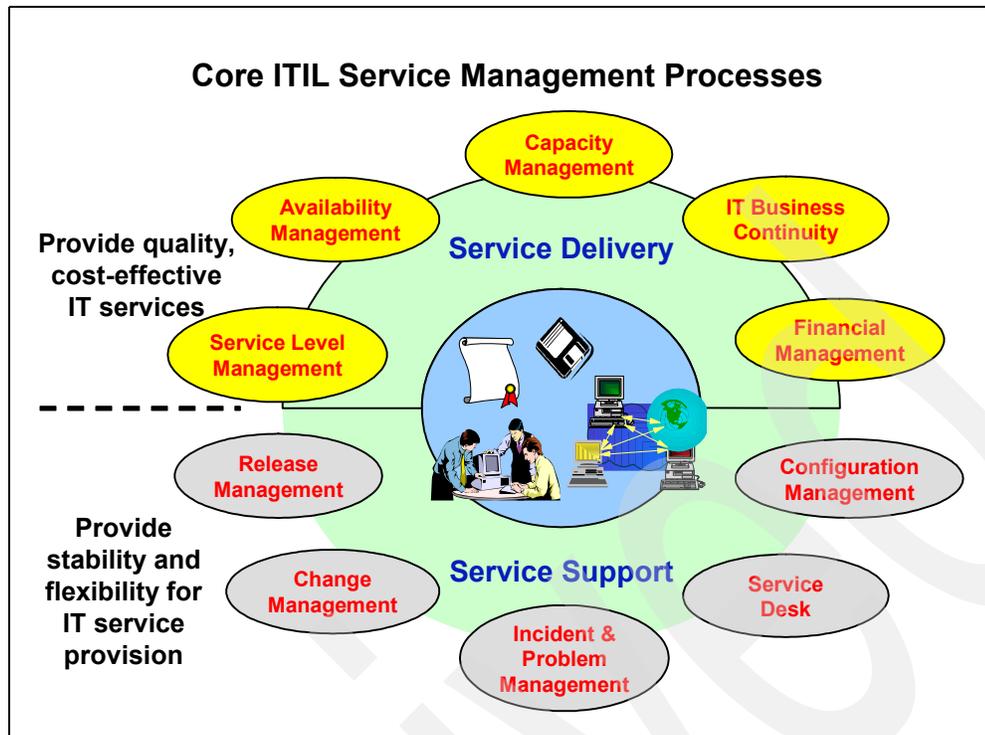


Figure 1-14 ITIL processes

Components of Service Support and Service Delivery

This section discusses the various components involved.

Service Support

The processes in the Service Support group are all concerned with providing stability and flexibility for the provisioning of IT Services.

Configuration Management

Configuration Management is responsible for registering all components in the IT service (including clients, contracts, SLAs, hardware and software components, and more) and maintain a repository of configurable attributes and relationships between the components.

Service Desk

The Service Desk acts as the main point-of-contact for the users requiring service.

Incident Management

Incident Management registers incidents, allocates severity, and coordinates the efforts of the support teams to ensure timely and correct resolution of problems. Escalation times are noted in the SLA and are as such agreed between the client and the IT department. Incident Management also provides statistics to Service Level Management to demonstrate the service levels achieved.

Problem Management

Problem Management implements and uses procedures to perform problem diagnosis and identify solutions that correct problems. It registers solutions in the configuration repository, and agrees on escalation times internally with Service Level Management during the SLA negotiation. It provides problem resolution statistics to support Service Level Management.

Change Management

Change Management ensures that the impact of a change to any component of a service is well known, and the implications regarding service level achievements are minimized. This includes changes to the SLA documents and the Service Catalog, as well as organizational changes and changes to hardware and software components.

Release Management

Release Management manages the master software repository and deploys software components of services. It deploys changes at the request of Change Management, and provides management reports on the deployment.

Service Delivery

The processes in the Service Delivery group are all concerned with providing quality, cost-effective IT services.

Service Level Management

The purpose of Service Level Management is to manage client expectations and negotiate Service Delivery Agreements. This involves finding out the client requirements and determining how these can best be met within the agreed budget. Service Level Management works together with all IT disciplines and departments to plan and ensure delivery of services. This involves setting measurable performance targets, monitoring performance, and taking action when targets are not met.

Financial Management for IT Services

Financial Management registers and maintains cost accounts related to the usage of IT services. It delivers cost statistics and reports to Service Level Management to assist in obtaining the right balance between service cost and delivery. It assists in pricing the services in the Service Catalogue and Service Level Agreements.

IT Service Continuity Management

Service Continuity Management plans and ensures the continuing delivery, or minimum outage, of the service by reducing the impact of disasters, emergencies, and major incidents. This work is done in close collaboration with the company's business continuity management, which is responsible for protecting all aspects of the company's business including IT.

Capacity Management

Capacity Management is responsible for planning and ensuring that adequate capacity with the expected performance characteristics is available to support the Service Delivery. It delivers capacity usage, performance, and workload management statistics, as well as trend analysis to Service Level Management.

Availability Management

Availability Management is responsible for planning and ensuring the overall availability of the services. It provides management information in the form of availability statistics, including security violations, to Service Level Management. This discipline might also include negotiating underpinning contracts with external suppliers, and a definition of maintenance windows and recovery times.

1.6.3 ITIL and ILM value

ILM is a service-based solution with policies and processes. The ITIL methodology has the processes required for delivery and support storage services to manage the lifecycle of information.

The ILM components tiered-storage, archive management, long-term retention, and data lifecycle management, aligned to ITIL processes, are a powerful solution for IT organizations to manage their data. By implementing ILM within the ITIL methodology, they are able to achieve its objectives, enabling the management of data lifecycle, and providing quality, stability, flexibility, and cost-effective IT services.

1.7 The technology layers of an ILM storage infrastructure

Information lifecycle management is not a hardware box or one or more software components, but rather a combination of multiple hardware and software components that interact based on predefined rules and processes to store data and information about the most effective and efficient infrastructure.

There are multiple aspects that drive ILM, such as cost, efficiency, and the requirement to manage risk and compliance. What do these aspects mean?

For example, cost is often a driver because the amount of data to manage keeps growing and we would like to store part of the data on more cost effective devices and not all on enterprise class disk storage. We can start thinking of moving less important data, but which data? Therefore, we require tools and processes to classify the data and assign it to the appropriate storage hardware tiers. This leads to many aspects related to the efficient use of storage and the classification, placement, movement, retention, and protection of data between tiers.

We simplify the complexity by breaking down the problem into three separate areas:

- ▶ Information management
- ▶ Storage management
- ▶ Storage

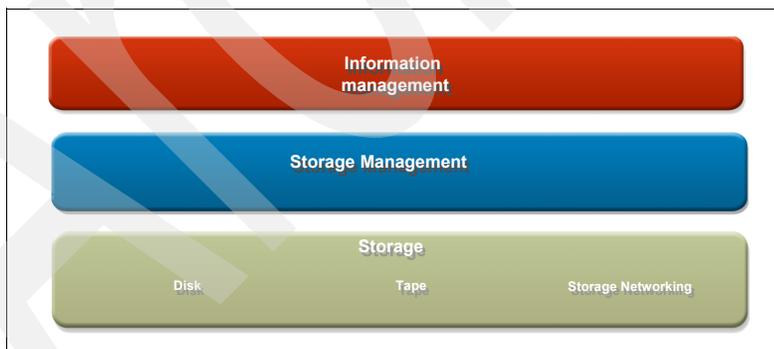


Figure 1-15 Technology layer

We discuss these three areas in greater detail, and we present them in reverse order because they present growing levels of complexity.

Storage infrastructures are relatively well understood, disk and tape devices and the like.

Storage management aspects are also quite widely understood. Storage has to be provisioned, protected, monitored, and data has to be copied or moved between storage devices for backup and archival reasons.

The information management layer is often less widely understood; the concepts and functions it provides might not be widely known. This layer is about classifying, retaining, indexing, and so on.

1.7.1 The storage hardware layer

The storage hardware layer comprises disk and tape storage devices, network attached storage systems, the DR550 data retention appliance and more. Virtualization plays an important role here; it is on the border between the storage and storage management layer, and offers, among other things, a simplified of the underlying storage infrastructure. Other solutions offer tape virtualization.

We illustrate some key products in this area in Part 2, “ILM building blocks” on page 41. In Chapter 5, “Tiers of storage” on page 111 we introduce the various hardware products and in Chapter 6, “IBM System Storage DR550” on page 141 we describe the DR550.

1.7.2 The storage management layer

The storage management layer (see Figure 1-16) offers functions to manage data for archival and retention, policy based data migration between storage tiers and data protection functions. These functions are offered by the IBM Tivoli Storage Manager family of products and by the IBM System Storage Archive Manager (SSAM) and DR550 retention solutions.

A second set of functions that are located in the storage management layer are relative to data and storage monitoring and data movement between the storage tiers. TotalStorage Productivity Center offers a comprehensive set of functions in this area and, in particular, functions to analyze and monitor storage usage and perform manual or automated actions based on predefined usage patterns or usage exceptions. Policies can be to periodically checked for specific file types on a file server and then migrate files that fall into this category onto a different storage tier.

We illustrate key components in this area in Chapter 5, “Tiers of storage” on page 111. We introduce IBM Tivoli Storage Manager and its sister product SSAM, which are the software components in the DR550 and are fundamental components in most of the ILM solutions discussed. Also, in Part 3, “Strategies and solutions” on page 157 we describe the use of IBM Tivoli Storage Manager as part of various ILM solutions.

We do not discuss the TotalStorage Productivity Center (TPC) software product in this book. For information about TPC and its interactions with IBM Tivoli Storage Manager, refer to the IBM Redbook titled *ILM Library: Techniques with Tivoli Storage and IBM TotalStorage Products*, SG24-7030, which is available for download at:

<http://w3.itso.ibm.com/itsoapps/Redbooks.nsf/RedbookAbstracts/sg246490.html?Open>



Figure 1-16 Storage Management Layer

1.7.3 The information management middleware layer

The first layer is information management middleware. Business applications rely on data, data contained in databases, local or remote filesystems, data received from external sources and stored locally. Often most of the data resides on disk with perhaps some small part of it located on tape or optical devices.

The information management layer concentrates on managing the data from an application's point of view. Conceptually the information management layer receives the data from an application and offers services such as storage and retrieval, archiving and retention, indexing and search, ingestion and distribution. For example, it could assist in storing the data on the appropriate device at different moments in its lifecycle, retaining it as required and making it searchable and available to the users.

The information management layer offers data services to applications. What kind of data service varies with the application's requirements. For example, an information management for a mail product might offer offline mail attachment storage to reduce the size of mail servers, while another product might offer functions such as retention and search capabilities.

There are many products in this area, products such as IBM DB2 Content Manager and IBM DB2 Commonstore for applications such as SAP, Exchange, and Domino.

We discuss the information management middleware in more detail in Chapter 3, "Information Management software" on page 43. We introduce various IBM information management products, and in Part 3, "Strategies and solutions" on page 157, you can see how many of these products fit into specific solution areas such as e-mail management and database archiving.

Archived



Planning for ILM

In this chapter we describe an approach to developing an Information Lifecycle Management (ILM) strategy, based on business requirements and illustrating possible trade-offs. We also discuss the diverse and sometimes conflicting requirements that guide and condition the solution, with particular attention to aspects pertaining to compliance with legal requirements.

We cover the following topics:

- ▶ Business drivers: cost reduction and simplification; improvement of efficiency; managing risk; and streamlining compliance
- ▶ The focus areas of information management and tiered storage
- ▶ Taxonomy of legal requirements and possible technological solutions

2.1 Business drivers: cost and efficiency

In this section we consider what is driving the requirement for ILM solutions. We cover some aspects that are very important when defining the correct approach to your ILM solution.

2.1.1 Challenges

Today many installations are facing information management and storage challenges due to the volume of data and complexity of the environment. Some recurring problem themes and reasons for the concentration on storage management are as follows:

- ▶ Surge in criticality, value, and volume of data:
 - Data being projected to grow at an annual rate of 64%
 - Outpacing the ability of IT to collect, store, and manage it by traditional means
- ▶ Excessive storage costs and missed service level objectives
- ▶ Compliance with regulatory requirements and audit procedures
- ▶ Ability to effectively access and gain insight from information after it has been stored

These challenges are impacting the ability to optimize information value and deploy Information On Demand solutions, as outlined in Figure 2-1.

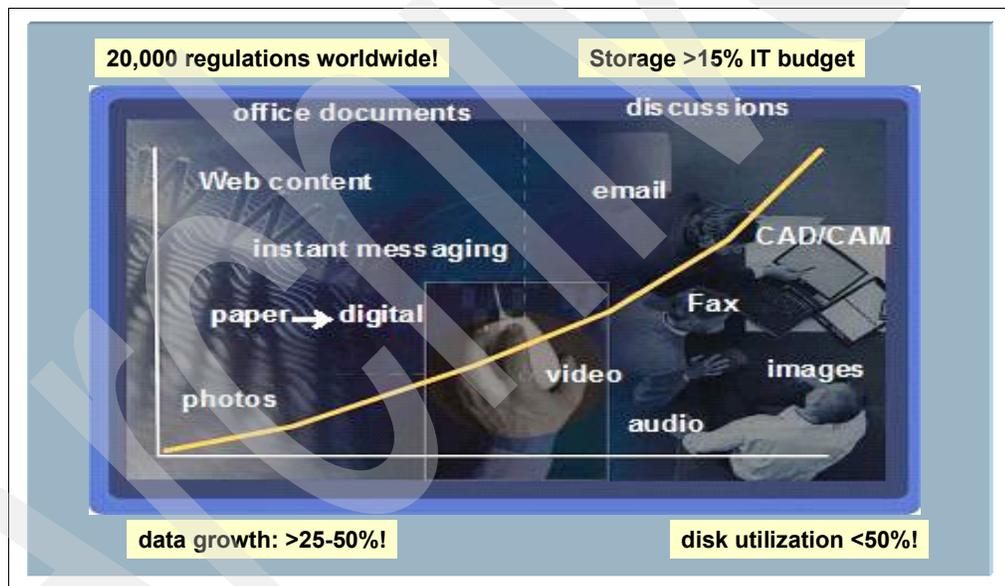


Figure 2-1 Information management and storage challenges

Next we discuss the main challenges and focus areas that installations are facing today, when trying to improve information management in their storage environments. It is not enough to consider only the cost of the technology — you must also decide which approach is best suited for you.

There are four typical reasons that we hear from installations regarding why they must improve their information management and storage:

- ▶ A surge in volume of data:

Many installations are experiencing a surge in data — that is, the criticality of that data, the value of that data, and the volume of that data. Installations are starting to think a lot about what is happening with all this data.

There seem to be two main reasons why data is growing at such a significant rate:

- One reason is that installations are adding new business, new volumes, new applications, and new users. These are all very *good reasons*. Interestingly enough, external research from META Gartner suggests that this might be 35 to 40% of the reason why data grows.
- The second major reason why data grows is because installations have inefficient or non-existent policies in place. They are making a lot of copies of copies. They have no control on the demand for storage. These are the so-called *bad reasons*.

Our experience shows that in many installations this is a very important reason why they are experiencing a huge growth in data. It is important to differentiate these two reasons because there are probably different strategies would want to take. Therefore, to summarize why installations might want to improve their information management and storage, we can simply say that their data is growing at an alarming rate.

▶ Excessive costs and missed service:

The second major reason is that many installations find that their costs are rising and they are not able to meet service objectives, even though they are spending a lot more money. It is interesting to explore this aspect in more detail, because it turns out that installations are spending more money on the process organization and the governance aspect, such as storage management, more so than on the procurement of hardware and software.

Some installations realize this and some do not. It is important to understand this aspect, because we are looking for strategies to enable installations to address the storage issues. However, if the installation is only spending on the technology component, they might not get the expected results.

▶ Compliance and regulatory requirements:

A third driver that is causing installations to focus on improving information management and storage is centered around compliance. Very often, that is because across industries, there are a variety of regulatory requirements, governmental regulations, and audit procedures requiring them to understand:

- What data they have
- How long they must keep it
- What they must do with it

Therefore, in their attempts to comply with governmental regulations, installations are having to improve their information management and storage strategies.

▶ Effective data access to gain insight:

The fourth reason why installations want better information management and storage has to do with the fact that ultimately, they want make sure they can effectively access and gain insight from their information after they store it. This is really all about helping them to transform their data and information so it can be used to take action and make decisions.

In developing an ILM solution for an installation, it is important that you understand their priorities; this allows you to address their most urgent issues, for example:

- ▶ Is it a compliance problem, or is it a surge in criticality and volume?
- ▶ Is effective data access an issue?

In the following sections we describe various solutions to address the different problem areas.

Some storage facts

Figure 2-1 on page 28 shows some very interesting statistics about the storage environment:

- ▶ Storage accounts for 15% or more of total IT storage budgets. Therefore, installations are spending a lot on storage and consequently paying a lot of attention to it.

- ▶ Data growth is rapidly rising, estimated at over 50% annually. The average fortune 500 company is running close to 150 TB of storage by now and some industries such as health care and life sciences are growing their data at one TB a day.
- ▶ We also find that the utilization of disk that installations have in their environment is low, often less than 50%. Therefore, there is a large degree of inefficiency in this area.
- ▶ There are also many regulations across different industries and countries around the world that are causing installations to focus on compliance related aspects.

Explosive data growth coupled with years of decentralized IT management practices has allowed storage environments to grow out of control, they have evolved into expensive, complex systems with fragmented data and legacy management processes. IBM Information Lifecycle Management solutions are designed to help installations effectively manage and store their information over its lifecycle, based on its value to their business operations.

2.1.2 The fluctuating value of data

Because not all data is created equal and the value of that data to business operations fluctuates over time, as illustrated in Figure 2-2, many installations are reevaluating their information management strategies.

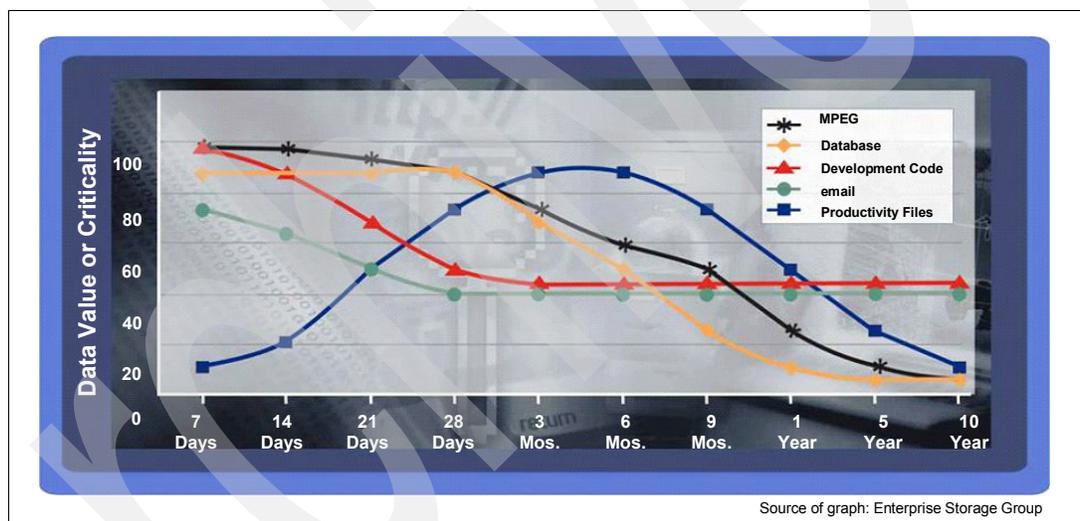


Figure 2-2 Fluctuating value of data

Installations can no longer afford to treat all data the same, they require the ability to align their IT investment with the true information value. The basic message in this chart is: *Do not treat all your data the same way.*

The first aspect to consider is that there are a variety of data types that installations have in their environment. In our example we have data such as MPEG, database, development code, e-mail, and productivity files. You can use TotalStorage Productivity Center (TPC) for data as a way to gather an inventory data and get an understanding about what different types of data exist in your environment.

If you look at the y-axis on the chart, you see data value or criticality. This is a relative number, different data types might have different values to the business. The way to understand the relative value of data types is by collaborating with a business owner or somebody who really knows the data so that they can help us understand what the particular value of the data is: different data types have different values.

Interestingly enough, as you look across the x-axis, with the passing of time you can see that the business value for each data type tends to fluctuate. There are different patterns that emerge, increasing and decreasing, but eventually, the value declines over time.

All of this leads us to the conclusion that if you have different data types and they have different values and each different value fluctuates over time, do not treat all your data the same way by having the same service level or using one expensive tier of storage. It just does not make sense from an efficiency point of view. And that is really the major conclusion about this chart. We can leverage IBM TotalStorage Productivity Center for data and some analysis to help us construct this picture for a specific installation.

Here is one other hint: Where the data fluctuates and the value goes down, this provides an artistic (rather than scientific) view showing where you can move data to a lower cost tier of storage and a lower service level.

Therefore, information is not static; its value changes during its lifecycle. As it ages, or passes specific business events such as the conclusion of a contract cycle, you might want to manage it appropriately. Some information has to be replicated and stored on a high performance storage and infrastructure, whereas lower priority information such as data kept for compliance purposes can be off-loaded and stored on less expensive storage mediums such as tape.

2.1.3 Objectives

Installations typically define specific objectives to support and improve their information management and storage environments, these objectives can be outlined and grouped into three distinct areas: cost, efficiency, and compliance:

- ▶ Cost reduction and simplification:
 - Controlling demand for storage
 - Improving asset utilization
 - Reducing hardware / software / storage personnel costs
 - Reducing data migration effort
- ▶ Improving efficiency:
 - Maximizing and sustaining efficiency by improving the current people, processes, and technologies being utilized to deliver storage services to the business
 - Defining and implementing the appropriate storage strategy to address current and future business requirements
 - Enhancing systems/e-mail performance
 - Making better use of existing information
- ▶ Managing risk and streamlining compliance:
 - Reducing organizational risk
 - Complying with governmental regulations

These three areas illustrate the specific objectives that we hear from installations when they are trying to improve their information management and storage environments. These areas describe the results they are expecting from their initiatives.

Whatever initiative or objective the installation has, or the result it is looking for, gives us a good idea about where to start and what solutions to bring to the table — which combination of IBM hardware, software, and services can help the installation get the results it requires. In the following sections, we explore these three objectives in more detail.

Reducing cost and simplification

The first and most commonly mentioned aspect concerns reducing cost and simplifying the environment. Here we have several different alternatives for gaining those results — initiatives such as controlling the demand; improving asset utilization; reducing hardware, software, and storage personnel costs; and also reducing data migration efforts. Therefore, those are different methods and strategies that installations might want to enable or enact in order to reduce costs and simplify their environment.

Improving efficiency

A second objective or result that installations are expecting when trying to improve storage management and information management is that they are typically looking for a plan to improve their efficiency. The efficiency is often obtained by taking a broader view of the IT problem; it is not only related to cutting costs. There are various strategies one might employ in order to achieve the efficiency objective:

- ▶ The first strategy involves maximizing and sustaining efficiency by concentrating on the current people, process, and technologies.
- ▶ A second strategy for improving efficiency is concerned with current and future business requirements. This is an important aspect for installations to take into account.
- ▶ A third strategy that can help installations improve their efficiency and get maximum gains is enhancing systems and e-mail performance. We look at some solutions that can help.
- ▶ A fourth strategy that installations can use for improving efficiency is to ensure that they can make better use of their information. They want to make sure that they are managing the right information and can have it available as required, so that they can make better decisions and have insight into the use of that information.

Managing risk and compliance

The third major objective that installations mention is that they want to manage risk and streamline compliance. There are a couple of different methods that can help us to get there:

- ▶ One method is reducing organizational risk by ensuring that your important data is stored, available, and kept secure, and ensuring that the data has not been tampered with intentionally or unintentionally. Therefore, it really revolves around archiving and retention.
- ▶ A second method is making sure that you are complying with governmental regulations. Therefore, you really have to consider what those regulations are, ensure that you are meeting them, and also, perhaps put some policies in place to enforce that compliance.

2.2 Focus areas

To address the objectives discussed in 2.1.3, “Objectives” on page 31, many installations are deploying Information Lifecycle Management (ILM) solutions. Figure 2-3 illustrates the various areas or tiers that together make up an ILM solution.

Notice that ILM spans these broad areas or layers:

- ▶ Data and infrastructure technology
- ▶ Information management
- ▶ Business policies and processes

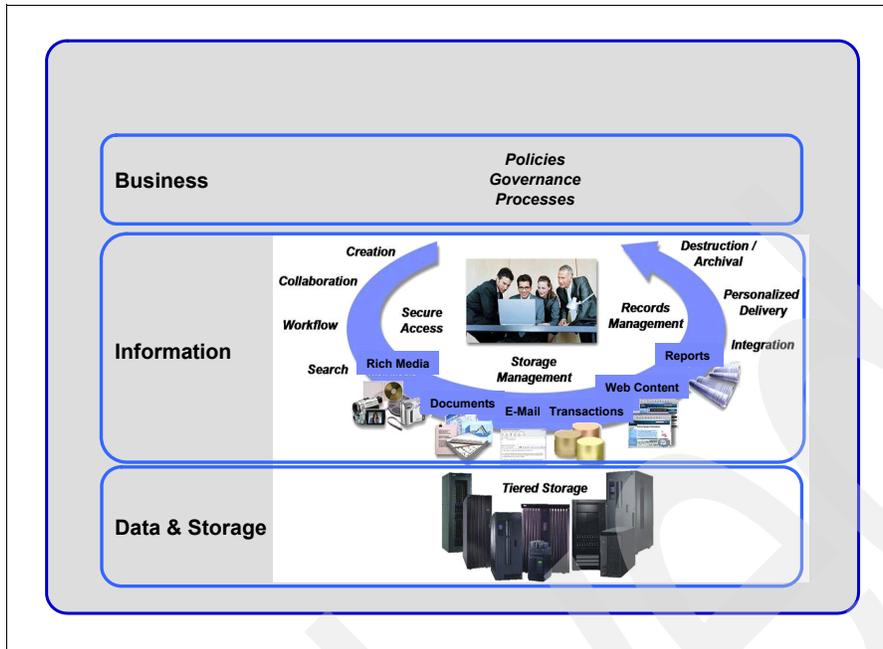


Figure 2-3 ILM solution areas or tiers

Figure 2-3 zeroes in on what Information Lifecycle Management actually is. In the following sections we discuss what Information Lifecycle Management is, how it can help installations, and why it is looked at as an answer to help address the objectives, which are reducing cost, improving efficiency, and managing compliance in their storage information environments.

The Storage Networking Industry Association (SNIA) defines ILM as follows:

“ILM is comprised of the policies, processes, practices, and tools used to align the business value of information with the most cost effective IT infrastructure from the time information is conceived through its final disposition. Information is aligned with business processes through management of service levels associated with applications, metadata, information, and data”.

We analyze the ILM definition from SNIA, the Storage Networking Industry Association which is made up of people, vendors, and organizations that collaborate together and IBM is one of the participants.

The first part of the definition talks about how “Information Lifecycle Management comprises policies, processes, practices, and tools.” From this, we can see that ILM is more than just a technology solution. Therefore, if all an installation is doing is focusing on the technology improvements, that probably is not going to get them the results they are looking for.

The second key aspect of the definition talks about “aligning business value of information with the most cost effective IT infrastructure”. It contains two very important statements:

- ▶ The first statement refers to the business value of information. That means we must collaborate with the user to understand what is the relative value of the different information types in the organization.
- ▶ The second statement refers to aligning that business value with some choices on the most cost efficient and cost effective IT infrastructure. Therefore, we want to make sure that for the most important business information, we are putting the most resources and effort behind managing it. And consequently we want to make sure that we are not wasting expensive space or expensive management time with applications that are not as critical or as crucial to the business, especially if it is at the cost of the most critical applications.

The third part of the definition states “from the time information is conceived through its final disposition”. The implication here that there are different relative values for that information as it goes through its lifecycle. There are many stages that it goes between. Therefore, this definition is very important.

The second sentence, “Information is aligned with business processes through management of service levels associated with applications, metadata, information, and data”, highlights one very important aspect, which is the management of service levels.

Again, this comes down to a choice of ensuring that we have got the most appropriate service level mapped to the most important information that comes from our business processes.

We can imagine an example considering two different applications: a Customer Relationship Management (CRM) application that is used for opportunity management processes, and a time reporting application. It probably would be a waste of money if we spent as much time and cost managing our time reporting application as we did for our CRM application. On the other hand, if we did not spend more money on the CRM application and only spent the minimal amount just to make sure that we are providing the same level of service that we do on our time reporting application, we probably would not be ensuring that we get the best return on investment for our most critical business applications.

ILM is not a product, but instead, ILM consists of the strategy, processes, and technology to effectively manage information through the phases of its lifecycle on the most cost effective IT Infrastructure.

ILM six best practices

Installations that drive and obtain the best results from ILM initiatives focus on the six best practices illustrated in Figure 2-4, showing a complete set of best practices recommended by IBM. These best practices represent a standard way or model to classify the various approaches that can be taken to create an ILM solution.

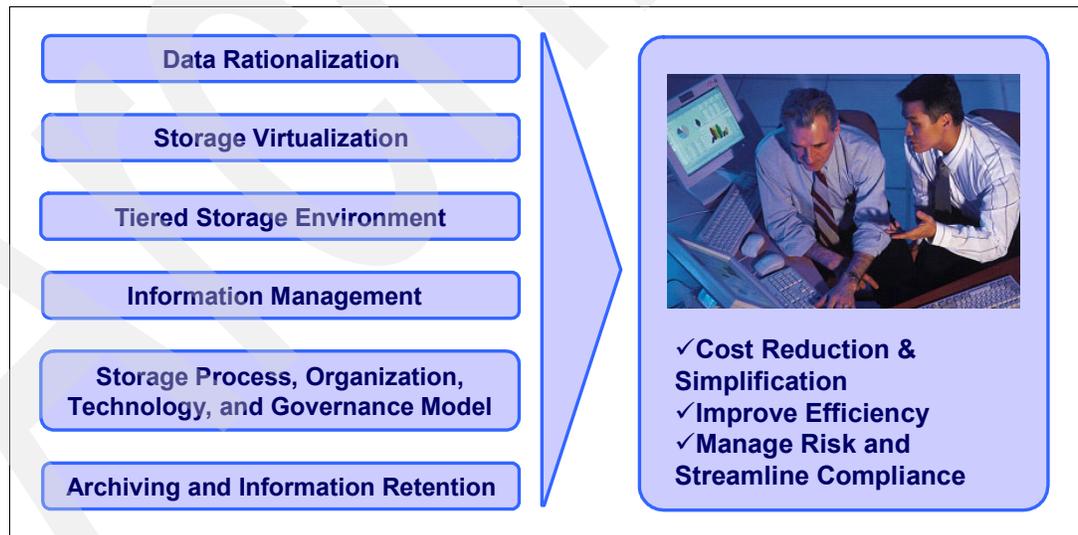


Figure 2-4 ILM initiatives six best practices

IBM has found that there are some patterns emerging here, and that the installations getting the best results from their ILM initiatives tend to focus on these six best practices.

Different installations can choose to concentrate on different areas, and in varying amounts. Some installations might want to start with one set of best practices first before considering others. There is a great degree of flexibility in the way things proceed, as installations try to reduce their cost, improve their efficiency, manage their risk, and streamline compliance. However, ultimately these six best practices can be expected to come together in some combination, to a greater or lesser extent. Therefore, let us take a closer look at them:

► **Data rationalization:**

The first best practice is data rationalization, where installations are separating their invalid data from their valid data. It concerns finding and classifying data, and determining what places the installation should be cleaning up because of duplicate, orphan, redundant, stale, or old data — which might be taking up space on expensive storage as well as requiring unnecessary and costly management.

► **Storage virtualization:**

The second best practice has to do with storage virtualization (Figure 2-5). There are a variety of uses for it, but at a high level, what virtualization does in the storage environment is to enable you to pool together different physical devices and present them in a logical fashion so that you are separating applications and users from the underlining physical storage and data. It allows for improved efficiency in managing the environment. It also allows for transparency to those users and applications so that you can change some underlining physical without disrupting the application and the users.

Different installations use virtualization in different ways — sometimes during data migration, sometimes as a general tool and technique across their storage environment, and sometimes to improve storage management.

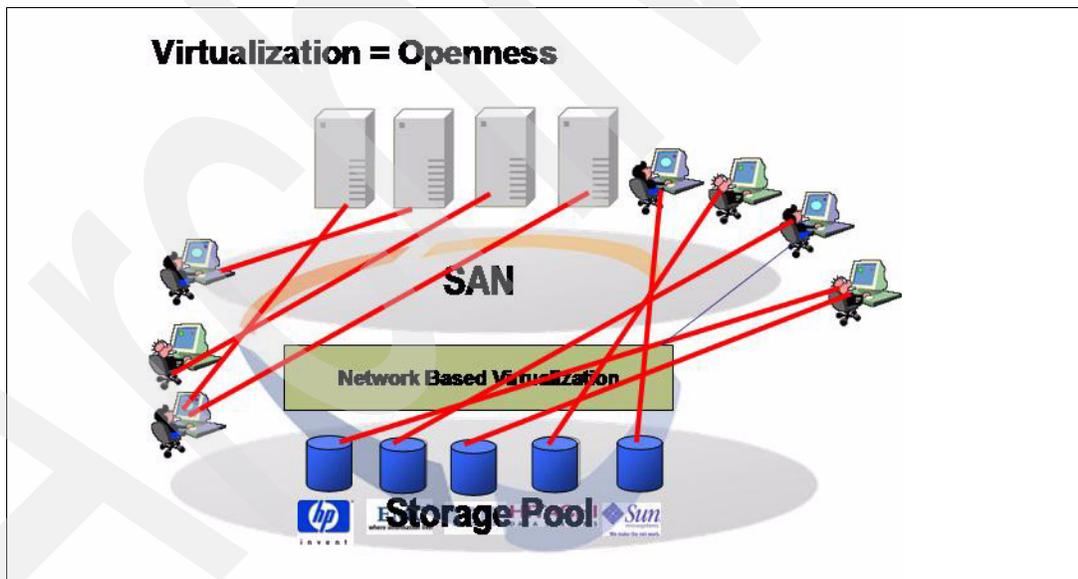


Figure 2-5 Storage virtualization

► **Tiered storage:**

A third best practice that we see is installations using tiered storage, which aligns variable cost hardware types with information classes, groups of data, and classes of service to create a variable cost storage environment.

In tiered storage, a very important aspect, installations are trying to figure out how they can leverage different variable cost technology types to support different requirements. Often the focus is on technological differences between the tiers. However, we have found it important to understand the differences between the tiers — basically, not just to use hardware characteristics, but really to use service level and business characteristics to help define the different service levels and the different tiers that are required in order to support the environment in a most efficient manner.

► Information management:

The fourth best practice, information management, refers to cases where installations are improving their data management, data access, and the insight gained by having the right information available when they require it. This can result in a higher business value.

► Storage process organization technology and governance:

The fifth best practice is centered around storage process organization, technology, and governance. These are very important areas for sustaining any improvements that might be gained through tiered storage, data rationalization, virtualization, and so on.

Very often this aspect is overlooked — we have found it important for installations to streamline their processes, roles, and responsibilities, to leverage a good integrated set of tools to manage the environment, and to make sure that they have a collaborative decision making model that is common across their business lines, as well as the appropriate infrastructure to ensure that they are getting efficiency out of the environment.

The focus on process organization and technology governance is a differentiator in helping to ensure that the results can be sustained. We have found that when installations do not use this best practice, they can become frustrated because they might gain some initial savings or some improvements in efficiency but these are never sustained over time.

► Archiving and information retention:

The sixth best practice entails archiving and information retention. This refers to two different aspects. The first aspect relates to helping installations improve performance by getting rid of the clutter of old data, whether it is e-mail, significant applications, or critical business applications. The second aspect relates to the compliance side of retention:

- What data do I have?
- How long must I keep it for?
- Why do I have to keep it around?
- How am I going to find it if I do decide to save it?

These are all important aspects revolving around Information Lifecycle Management and driving improvements towards cost reduction and simplification, improved efficiency, and managing risk and compliance.

2.3 Taxonomy of legal requirements

There is a rapidly growing class of data that is best described by the way in which it is managed rather than the arrangement of its bits. The most important attribute of this kind of data is its retention period, hence it is called *retention managed data* (Figure 2-6), and it is typically kept in an archive or a repository. In the past it has been variously known as *archive data*, fixed content data, reference data, unstructured data, and other terms implying its read-only nature. It is often measured in terabytes and is kept for long periods of time, sometimes forever.

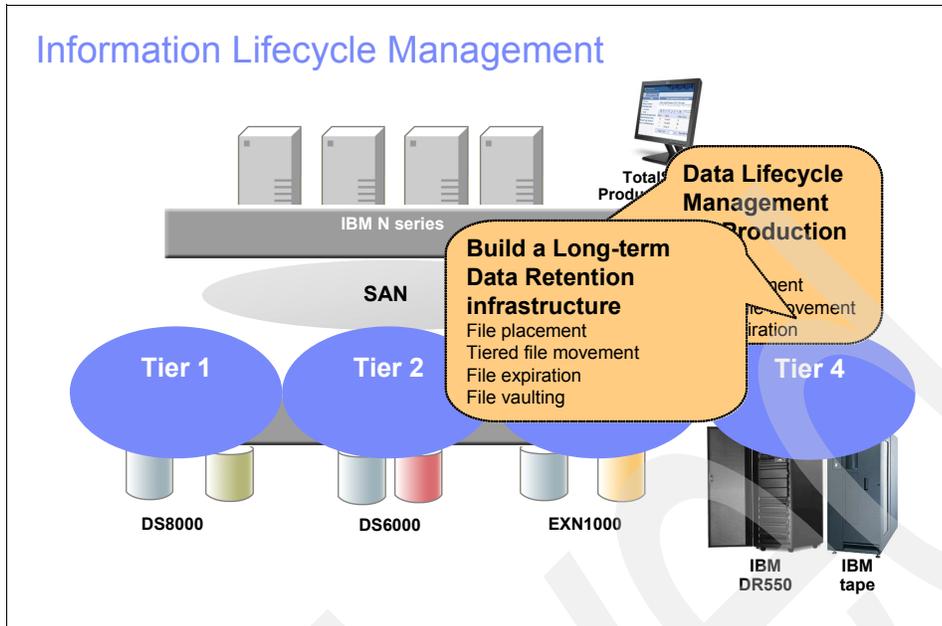


Figure 2-6 Data retention

In addition to the sheer growth of data, the laws and regulations governing the storage and secure retention of business and client information are increasingly becoming part of the business landscape, making data retention a major challenge to any institution. An example of these is the Sarbanes-Oxley Act of 2002 in the USA.

Businesses must comply with these laws and regulations. Regulated information can include e-mail, instant messages, business transactions, accounting records, contracts, or insurance claims processing, all of which can have different retention periods, for example, for 2 years, for 7 years, or retained forever. Moreover, some data must be kept just long enough and no longer. Indeed, content is an asset when it really must be kept. However, data kept past its mandated retention period could also become a liability. Furthermore, the retention period can change due to factors such as litigation. All these factors mandate tight coordination and the requirement for ILM.

Not only are there numerous state and governmental regulations that must be met for data storage, but there are also industry-specific and company-specific ones. And of course these regulations are constantly being updated and amended. Organizations have to develop a strategy to ensure that the correct information is kept for the correct period of time, and is readily accessible when it must be retrieved at the request of regulators or auditors.

It is easy to envision the exponential growth in data storage that results from these regulations and the accompanying requirement for a means of managing this data. Overall, the management and control of retention managed data is a significant challenge for the IT industry when taking into account factors such as cost, latency, bandwidth, integration, security, and privacy.

2.3.1 Regulation examples

It is not within the scope of this book to enumerate and explain the regulations in existence today. For illustration purposes only, we list some of the major regulations and accords in Table 2-1, summarizing their intent and applicability.

Table 2-1 Some regulations and accords affecting companies

Regulation	Intention	Applicability
SEC/NASD	Prevent securities fraud.	All financial institutions and companies regulated by the SEC
Sarbanes Oxley Act	Ensure accountability for public firms.	All public companies trading on a U.S. Exchange
HIPAA	Privacy and accountability for health care providers and insurers.	Health care providers and insurers, both human and veterinarian
Basel II aka The New Accord	Promote greater consistency in the way banks and banking regulators approach risk management across national borders.	Financial industry
21 CFR 11	Approval accountability.	FDA regulation of pharmaceutical and biotechnology companies

For example, in Table 2-2, we list some requirements found in SEC 17a-4 to which financial institutions and broker-dealers must comply. Information produced by these institutions, regarding solicitation and execution of trades and so on, is referred to as compliance data, a subset of retention-managed data.

Table 2-2 Some SEC/NASD requirements

Requirement	Met by
Capture all correspondence (unmodified) [17a-4(f)(3)(v)].	Capture incoming and outgoing e-mail before reaching users.
Store in non-rewritable, non-erasable format [17a-4(f)(2)(ii)(A)].	Write Once Read Many (WORM) storage of all e-mail, all documents.
Verify automatically recording integrity and accuracy [17a-4(f)(2)(ii)(B)].	Validated storage to magnetic, WORM.
Duplicate data and index storage [17a-4(f)(3)(iii)].	Mirrored or duplicate storage servers (copy pools).
Enforce retention periods on all stored data and indexes [17a-4(f)(3)(iv)(c)].	Structured records management.
Search/retrieve all stored data and indexes [17a-4(f)(2)(ii)(D)].	High-performance search retrieval.

2.3.2 IBM ILM data retention strategy

Regulations and other business imperatives, as we just briefly discussed, stress the necessity for an Information Lifecycle Management process and tools to be in place. The unique experience of IBM with the broad range of ILM technologies, and its broad portfolio of offerings and solutions, can help businesses address this particular requirement and provide them with the best solutions to manage their information throughout its lifecycle. IBM provides a comprehensive and open set of solutions to help.

IBM has products that provide content management, data retention management, and sophisticated storage management, along with the storage systems to house the data. To specifically help companies with their risk and compliance efforts, the IBM Risk and Compliance framework is another tool designed to illustrate the infrastructure capabilities required to help address the myriad of compliance requirements. Using the framework, organizations can standardize the use of common technologies to design and deploy a compliance architecture that can help them deal more effectively with compliance initiatives.

Some key products of IBM for data retention and compliance solutions are:

- ▶ IBM Tivoli Storage Manager, including IBM System Storage Archive Manager
- ▶ IBM DB2 Content Manager Family, which includes DB2 Content Manager, Content Manager OnDemand, CommonStore for Exchange Server, CommonStore for Lotus Domino, and CommonStore for SAP
- ▶ IBM System Storage N series
- ▶ IBM DB2 Records Manager
- ▶ IBM TotalStorage DS4000 with S-ATA disks
- ▶ IBM System Storage DR550
- ▶ IBM TotalStorage Tape (including WORM) products

For details on these products, see Part 2, “ILM building blocks” on page 41.

Important: The IBM offerings are intended to help clients address the numerous and complex issues relating to data retention in regulated and non-regulated business environments. Nevertheless, each client’s situation is unique, and laws, regulations, and business considerations impacting data retention policies and practices are constantly evolving. Clients remain responsible for ensuring that their information technology systems and data retention practices comply with applicable laws and regulations, and IBM encourages clients to seek appropriate legal counsel to ensure their compliance with those requirements. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the client is in compliance with any law.

2.4 Content management solutions

IBM offers a variety of products and technologies to effectively capture, manage, and distribute content that is important to the operation of your organization. IBM delivers an integrated content management portfolio that enables you to transact daily operations and collaborate across diverse applications, business processes, and geographic boundaries. See Figure 2-7.

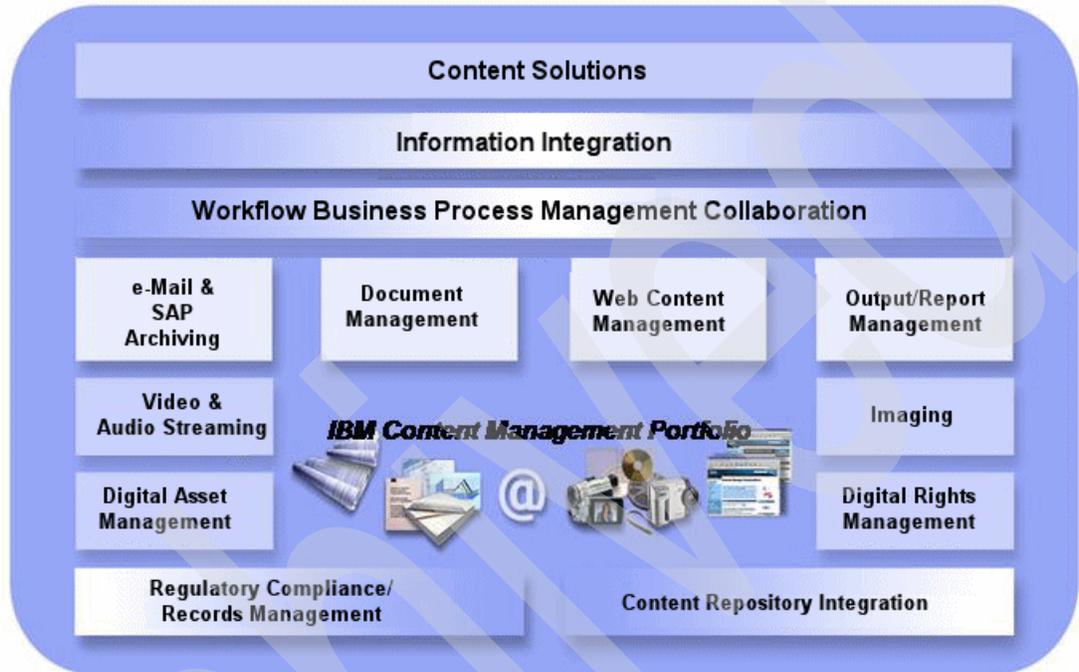


Figure 2-7 Content management

IBM has offerings supporting all information types, including images, documents, reports, e-mail, instant messaging, discussion forums, e-statements, audio, video, data, and Web content and integrates this information with your existing e-business applications. IBM capabilities can service requirements that range from workgroups to high volume business processes.

IBM offers a full range of content management functionality in its portfolio that can help you address your content management issues. You can start with the most important area for your business, perhaps it is document management, or Web content management, and then expand to other areas as required, all while leveraging common, open technologies. You can leverage all or part of the portfolio to solve a particular business problem, with it fitting into your existing operating environment. IBM not only helps solve today's business problems but provides a flexible infrastructure that can be extended into the future.

IBM can support your complete content management requirements, including support for all forms of information, document and records management, digital rights management, collaboration, workflow processing, and the integration of information from multiple source repositories. IBM is unique in that it offers the most comprehensive range of integrated capabilities, which include content, collaboration, process, information, and storage management, each best of class in their own right. Also, we have deep, global expertise in delivering content management solutions that are reliable, scalable, and secure.



Part 2

ILM building blocks

In this part of the book we discuss, in more detail, the building blocks to ILM:

- ▶ IBM Tivoli Storage Manager
- ▶ IMS™
- ▶ The IBM software and hardware suite of storage products
- ▶ Our retention management solution

Archived



Information Management software

IBM Content Management and Discovery software integrates and delivers critical business information that offers new business value, on demand. The software and solutions support multiple information types, such as images, documents, e-mail, Web content, e-records, and multimedia, and provide the appropriate content, based on user intent and relevancy. The IBM Content Management and Discovery portfolio is designed to help transform business with improved productivity and streamlined compliance.

This chapter describes Content Management in general and the Content Management and Discovery software products of IBM.

3.1 Content Management

Content Management transforms the way you do business. Content management software captures, stores, manages, integrates, and delivers all forms of digital content across a company's entire value chain from employees to customers to suppliers and partners — to create real business value. Content management systems and integrated processes provide the unified approach for managing multiple content types. IBM offers an open and completely integrated enterprise content management portfolio that supports all industries, regardless of company size, worldwide. See Figure 3-1.

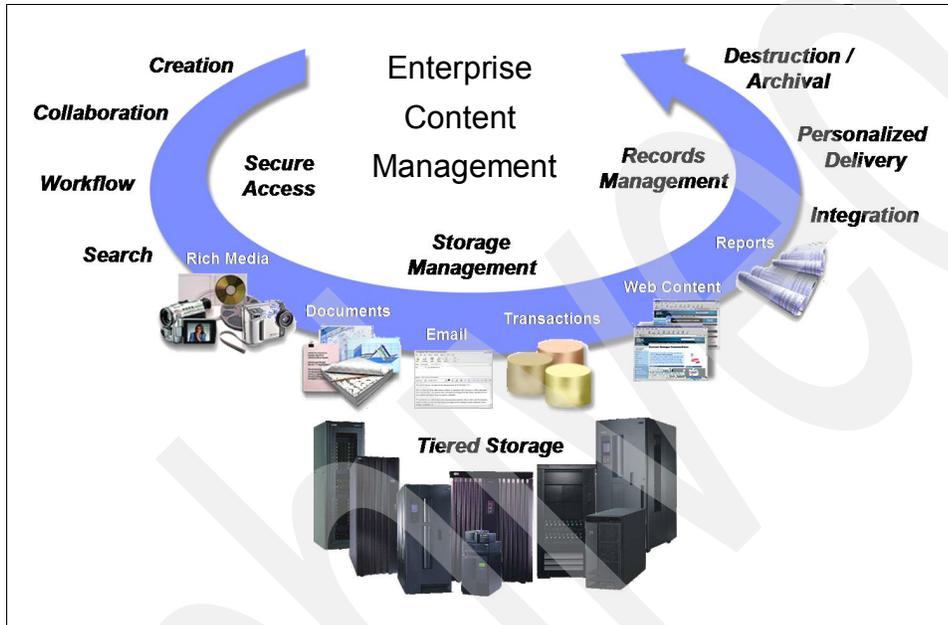


Figure 3-1 Enterprise content management

A definition of Enterprise Content Management is that it is a framework for creating, managing, integrating, Web enabling, and delivering unstructured digital content across the enterprise and beyond, to employees, customers, and trading partners, in a way that creates real business value.

It is also necessary for enabling and supporting an on demand business environment. But what is content? Figure 3-2 illustrates some common types of content.

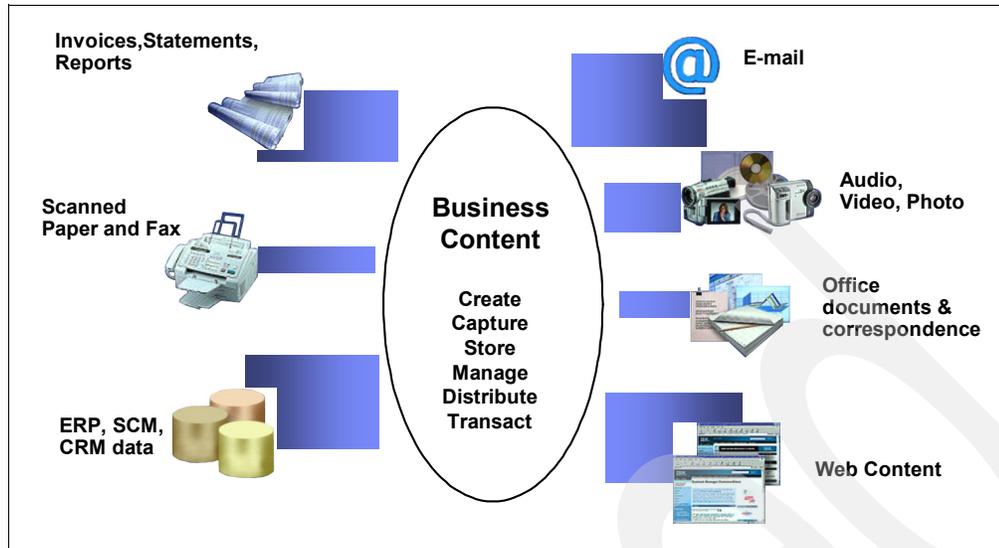


Figure 3-2 Types of content

Content can mean many different things, therefore, let us be sure that we all have a common understanding of what the scope of business content is. We start at the top of the diagram:

- ▶ Our computers create printable output such as invoices, statements, and reports. Today much of this data is either printed and put in file folders or stored on microfiche.
- ▶ Many documents enter a business as scanned paper or faxes.
- ▶ ERP, SCM, and CRM systems have a lot of data that must be archived in order to maintain manageable database sizes and system performance. These applications also have associated content such as invoices that support an Accounts Payable process.
- ▶ Today, e-mail has evolved from being a collaborative internal tool and has now become a vital part of communications to customers, agents, brokers, and partners. Therefore, this e-mail now contains potential business commitments and is critical business content that must be saved, often for reasons of legal compliance.
- ▶ Audio and video are becoming more and more important to the business. Some examples are audio or video conferences, online learning materials, videos related to other customer content, and audio statements.
- ▶ Many employees have to create office productivity documents and spreadsheets in the course of doing business. This also becomes part of the business content.
- ▶ Finally, there is Web content, which includes all of the graphic files, text components, and animation that are increasingly important as business is extended to the Web.

These are some of the content issues found in business. There is so much paper that it clogs work processes. It is difficult to find what is required in a timely manner, and folders have become inches thick, with no table of contents to aid the search. Productivity is impacted because information is not instantly available. In addition, work processes are not automated. We have done a good job of automation on our business systems, but the manual inefficient work processes are still in place. Work is still hand delivered from one desk to another, slowing down business process cycle times. There is also no concurrent use — we cannot run parallel processes, unless someone makes copies of the material.

Today e-mail can be discarded by users with no business controls at all, although it forms vital business assets that can potentially help us to avoid litigation, as well as to meet company and regulatory requirements. As mentioned before, office documents are not in an enterprise

library, and are not necessarily in the customer folder where they should be unless someone took the time to print them. Documents that should be shared are on one person's computer.

Overall customer service is slowed because information is not easily accessible while you are on the phone with the customer. This results in many callbacks, with costly delays that are irritating to customers. And finally, it is difficult for users to contribute Web content directly. That requires a Web master, who becomes a bottleneck in the process. Therefore, current content is delayed in getting to the site.

In summary, there are many issues related to the business content that we handle every day. Figure 3-3 illustrates the various aspects of content management.

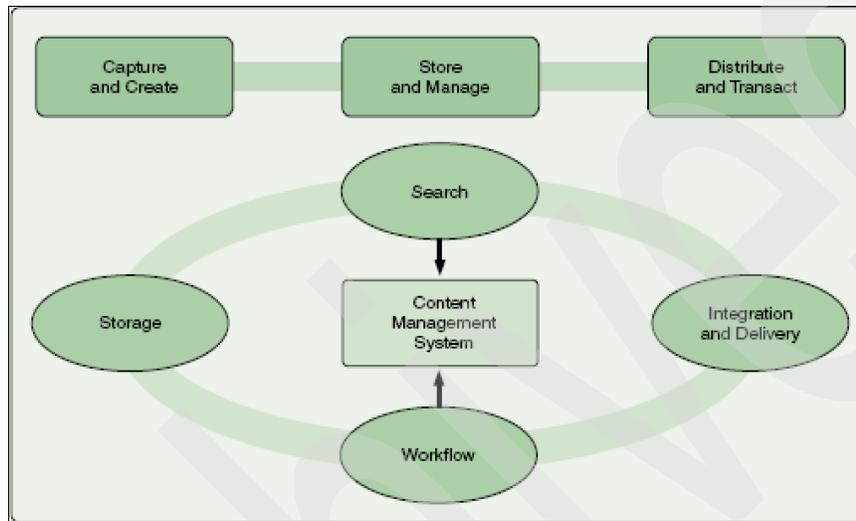


Figure 3-3 Content management objectives

Let us look at the fundamental elements that make up a robust content management system. Essentially, a system has to support all three stages of an information or content *value chain*, from the *creation and capture* of content in a digital format to the *management* of content in repositories that provide storage, archiving, workflow, search, and access capabilities, to the ultimate *delivery* of content through any sort of device.

All three stages, regardless of content type or industry, require a robust, scalable infrastructure that can support numerous transactions through a variety of channels such as applications, portals, browsers, kiosks, telephones, and mobile devices. The system must be able to store any amount of content, enable secure and reliable distribution, handle unpredictable loads, and provide a foundation for future growth.

Historically, content management systems have been implemented to address specific, defined tasks, such as insurance claims processing or capturing employee performance information. In these cases, the pool of content users was well defined and often contained within a single department. However, as we move into the on demand era, an increasing number of users, within and beyond the enterprise, require access to content, increasing the user base from hundreds of users to thousands.

To meet these requirements in a cohesive and coherent way, you must have a platform that provides repository services such as capture, creation, organization, workflow, and archival of content. It should deliver a consistent information model with transaction security, so that enterprise content solutions can focus on getting the right information to the right people at the right time, without having to worry about managing the underlying infrastructure.

Lifecycle management, which is managing the document from creation through to deletion at the appropriate time, is more frequently becoming a driver as organizations struggle to deal with regulatory pressures from government and industry bodies.

In the following sections, we examine each of the three stages in the value chain in detail.

3.1.1 Creation and capture of content

The first stage involves capturing content in a digital form. Some content is *born* digital and is relatively easy to store in a content management system. This would include presentations, documents created with word processing systems, digital photos, and Web pages.

Other forms of content, such as paper documents or video files, must be digitized through a scanning process. There are many effective solutions available today to help organizations get all of their content into a digital form so that it can be processed and managed through a content management system. However the content is created, transforming it into a digital object and storing it in a content repository is the first important step.

Here are some examples of IBM products and offerings in this area:

- ▶ IBM Workplace™ Forms replaces paper with secure XML forms capability.
- ▶ IBM Document Manager and its integration into Microsoft Office gives users advanced document management capabilities and the possibility to store this information directly into the centralized content repository.
- ▶ Partner solutions, such as those from Kofax, provide sophisticated scanning and image capture capabilities integrated with IBM content repositories.
- ▶ Other partners provide industry-specific capture capabilities such as video and multimedia capture and indexing, and integration with medical systems.
- ▶ Computer output can be captured, managed, and archived with Content Manager OnDemand.

3.1.2 Management of content

The second stage involves managing the content. A content management system might have to help move the content from person to person so that they can act on it, provide controls for restricting content access to only those who must work with it, keep track of what has happened to the content throughout its digital life, and provide the means for storing and disposing of the content when it is no longer required.

Whether it is managing Web images, e-mail, videos, or other documents, a content management solution should feature:

- ▶ A robust repository
- ▶ An open architecture
- ▶ Options for integration with applications
- ▶ Controls for managing the lifecycle of the information

Content management platforms must be scalable, robust, reliable, and secure to address the peaks in usage that come from a variety of users both inside and outside the corporate firewall. While some repetitive business applications have predictable access patterns, other applications, particularly those that provide customer's with access to information, might not be so predictable. A content management platform should scale from small departmental solutions to enterprise wide applications used by thousands of employees, as well as customer-facing e-business Web sites receiving millions of hits per day.

Beyond scalability characteristics, a content management platform should be based on open standards and support leading server platforms, database management systems, and packaged business applications.

By enabling you to use your current IT investments in software and hardware, an open architecture enables the flexibility to integrate content to any application and allow the seamless movement of content between organizations, customers, partners, and employees.

Finally, a robust content management system provides the capability to manage digital assets throughout their lifecycle, from creation to disposition. And the system should provide the flexibility for you to define the policies you want to implement for retaining and disposing of content in accordance with company policy and industry regulations.

3.1.3 Delivery of content

In the third stage, content management platforms must support the delivery of secure, personalized content through a variety of delivery channels, from portals to unique applications, as well as through a range of mobile and office-bound devices.

The necessity of handling different types of media across a whole range of business applications, including Enterprise Resource Planning, supply chain, and Customer Relationship Management applications, is increasing. These applications are being called on to handle content transparently in the application user interface. Systems such as SAP and Siebel® have the requirement to include images, Web content, scanned documents, and other types of information and to deliver the information about a variety of devices, all involving the transformation of the information to fit the form of the device.

Your content management system must be able to handle the transformation and delivery of information to meet the user's requirements and to display it in the preferred form.

3.2 Choosing the right product for content repository

IBM DB2 Content Manager and IBM DB2 Content Manager OnDemand are both content repositories. Deciding which of these products to use, or whether to use both products in parallel, depends on the business requirements.

In the following two sections we describe the differences between these products, and explain for which business cases they are primarily designed.

3.2.1 IBM DB2 Content Manager

Content Manager provides an open and comprehensive platform for managing all types of digitized content. It is available in a variety of platforms including Windows®, AIX®, Linux®, Solaris™, and z/OS, and supports DB2 or Oracle® as a database system.

DB2 Content Manager is built upon a relational database as stored procedures, leveraging the content retrieval and security, using IBM WebSphere® Application Server and object migration together with backup and recovery, using IBM Tivoli Storage Manager.

One can think of DB2 Content Manager as a central repository, in much the same way that you would use DB2 Universal Database™. However, Content Manager is a solution that enables workflow (both document centric and non-document centric), check-in/check-out, versioning of documents, finer access control, and privilege control for users and groups.

Support with Lightweight Directory Access Protocol (LDAP), fax solutions, scanning solutions, object migration and backup, and recovery are provided either out of the box or seamlessly integrated with third-party solutions. It serves as a place where you can store all sorts of documents, as well as retrieving, modifying, and archiving them for long term purposes.

Content Manager uses a triangular architecture, as shown in Figure 3-4. Client applications (running either in end-user desktops or mid-tier application servers) use a single object-oriented API to invoke all Content Manager services that are divided between a library server and one or more resource managers. The library server manages the content metadata and is responsible for access control to all of the content, interfacing with one or more resource managers. Resource managers manage the content objects themselves. Both the library server and resource manager can utilize LDAP services for user management and access control.

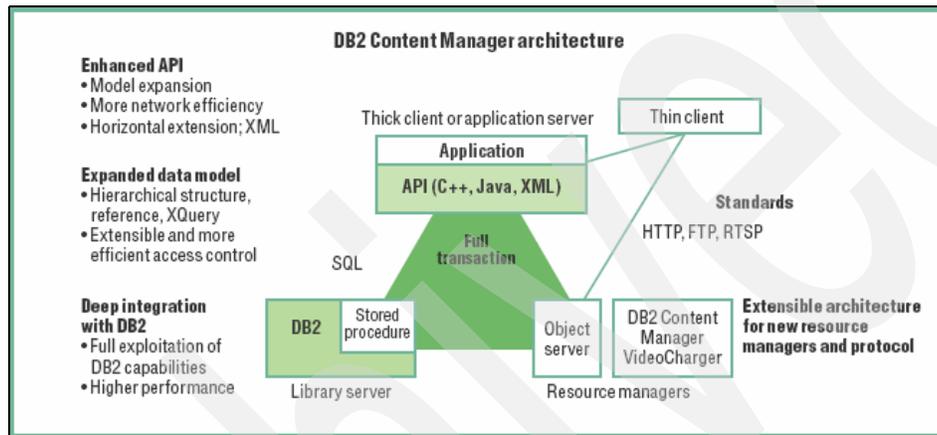


Figure 3-4 Components in Content Manager triangular architecture

Therefore, Content Manager is mainly the content repository for *inbound documents* such as scanned images, office documents, and e-mails. You can manage these documents with the help of the document management services of Content Manager. The folder management allows you to have a structured view and access to information with different hierarchical levels. An example might be a view of all hospital records with subfolders to each patient listed in the hospital or department of the hospital. The system lets you store and retrieve all kind of information related to an patient, such as X-ray radiographs (image), e-mails, doctors diagnosis (text), and so on.

Note: DB2 Content Manager does not provide any capturing, management, archiving and retrieval of computer output like DB2 Content Manager OnDemand does.

Document routing

Content Manager document routing provides the integrated capability to route work along a predefined process. A process defines the way users perform the work and the route through which work progresses. Different routing alternatives include:

- ▶ Sequential routing
- ▶ Parallel routing
- ▶ Branching based routing based on specified action or process values
- ▶ Collection points
- ▶ Server exits on entering/leaving work nodes
- ▶ Decision points
- ▶ User -defined actions/action lists

Graphical workflow builder is delivered with Content Manager to administer document routing and workflow functions. It enables the construction of workflow using GUI drag and drop. The decision point supports conditional branching based on the criteria defined. An action list is a list of actions defined for user applications. Parallel routing allows work packages to move along multiple routes in parallel. Line of business facilitates the integration of external business applications with a workflow. A subprocess helps the business analyst manage the complexity of a larger workflow through the reuse of existing processes.

An administrator can define the work process for a document's routing to model a specific business process step by step. After a work process has been defined, you can route documents through a work process that assigns items to individuals or groups for processing until the item's entire process has been completed. DB2 Content Manager document routing is integrated with access-control checking, user management and general system management to facilitate high quality document management and processing in a business environment. DB2 Content Manager also provides a consistent and repeatable process for document handling, so that you can enforce enterprise business rules.

Because DB2 Content Manager defines a formal process to model a business procedure, you can enforce business rules to help ensure the integrity of your business process. While document routing in DB2 Content Manager supports predefined and repeatable work processes, it also supports simple but dynamic routing, where users can make decisions at each step to continue to the next defined workflow path or move the work item to another path or work node.

Version management

You can store multiple versions of documents and parts within documents in DB2 Content Manager. DB2 Content Manager can create a new version when any changes occur in the document content or in its indexing attributes. Each version of a document is stored as a separate item in the system. Users can access the latest version or any version of the document by specifying the desired version number. By default, the most recent version is presented to the user, who can see if other versions exist. To limit the number of versions managed in the system, administrators configure how many versions exist for a single item. DB2 Content Manager automatically deletes older versions exceeding the limit.

The system administrator can determine, by item type, whether a store or update operation creates a version, modifies the latest version, or prompts the user to create a version.

Search and access

For a content management system to become effective and to enhance the productivity of its users in the day-to-day e-business environment, efficient search and access technologies play vital roles. DB2 Content Manager provides advanced search and access technologies that give you the power to locate and retrieve content for their business requirements quickly and accurately.

DB2 Content Manager uses three search methods, which are parametric search, full-text search, and combined parametric and full-text search:

- ▶ Parametric search lets you locate the contents by specifying criteria based on business metadata attributes such as customer or account numbers.
- ▶ Full-text search lets you enter free text or keywords as search criteria against text-indexed documents to locate documents that contain pertinent content anywhere within the body of the document.
- ▶ Combined parametric and full-text search allows you to enter both metadata attributes and full-text or keywords to expand search criteria.

DB2 Content Manager automatically indexes documents for subsequent full-text searching and adds it to the full-text index if this feature is configured for this item type.

Security and authentication

Authentication and authorization are critical when enterprise information assets are involved. For this reason, the DB2 Content Manager includes a sophisticated access control mechanism. Different users or group members can have different access rights to classes of content, individual folders, documents, or parts of documents. For example, a human resources application can allow an employee to see parts of his or her own personnel file, but limit access for some sections to that employee's manager, and other sections to human resources managers only.

The resource manager is the repository for objects stored in the DB2 Content Manager system. A single library server can support multiple local or remote resource managers, or a mixture of both. Users store and retrieve content in the resource manager by first submitting requests through the controlling library server. The library server validates the access rights of the requesting client, and then authorizes the client to directly access the designated resource manager to store or retrieve the objects.

Through LDAP integration support, DB2 Content Manager applications can take advantage of centralized users, groups, and server directories. DB2 Content Manager can be configured during installation (or later by using the system administration client) to communicate with an LDAP server. The LDAP server can manage user IDs and user groups with the information, then import it into DB2 Content Manager through either the system administration client or a provided import utility. DB2 Content Manager stores this information, giving the system speed and rich security features. DB2 Content Manager allows authorization checks at any level. When there are changes, this utility can keep users and groups synchronized between LDAP and DB2 Content Manager servers.

DB2 Content Manager supports the creation of administrative domains in the library server exclusive to a group of users. Each domain has one or more administrators to manage user access within that domain. Then, by default, users within each domain have access only to documents created within their domains. Administrative domains streamline and distribute the user management in a DB2 Content Manager configuration with a large user base divided among many departments.

For example, an insurance company could divide the DB2 Content Manager user administration by department, because users in the claims department do not have to view or work with any documents in the sales department. A central administrator can still view documents across domains by using appropriate access controls and views. Administrative domains are also particularly valuable to application service providers who manage large DB2 Content Manager facilities for more than one corporate client. One administrative domain can be created for all users belonging to one corporate client. This makes for a division and safeguard among different companies in the single DB2 Content Manager environment.

Storage management

The resource manager provides hierarchical storage management by working in conjunction with IBM Tivoli Storage Manager. When objects are first stored in a resource manager, they are assigned to a storage class and the associated storage system. Migration policies can be defined to automatically move objects from one storage class to another based on the duration of the object in a storage class. For example, objects that have been loaded onto the attached magnetic storage system for more than six months can be migrated to an optical disc or a tape for long-term archival to reduce storage costs.

Logging facilities

DB2 Content Manager provides an audit trail and logging facility with the ability to capture more detail for audit or charge-back billing purposes. This feature allows the administrator to define different levels of logging to capture functions performed on certain documents or folders by users. The logging facility captures user ID, time stamp, process ID, work basket or node ID (both from and to), suspend and resume times for each action, and an event code for each of the possible actions. It also logs administrator operations and stores the logged data in DB2 Universal Database tables. The administrator can use standard SQL reporting tools to create reports based on the captured data. He can audit users of the system, feed billing programs with usage statistics, and better understand how work moves through business processes.

Web services and XML

DB2 Content Manager includes Web services to deliver remote access to DB2 Content Manager functionality. It provides a Web services interface, that you can use within your applications, with other Web services interfaces, or in complex business processes to seamlessly perform actions against a DB2 Content Manager system regardless of the programming language they were written in or the platform on which they run.

The DB2 Content Manager Web service is a messaging-based communication model that defines loosely coupled and document-driven communication as illustrated in Figure 3-5. The client service requester invokes the Web service by sending it a complete XML document, in an SOAP message, which represents a particular request for a DB2 Content Manager operation, such as Search. The DB2 Content Manager Web service provider receives the XML document, processes it, and returns a message, as an XML document in another SOAP message.

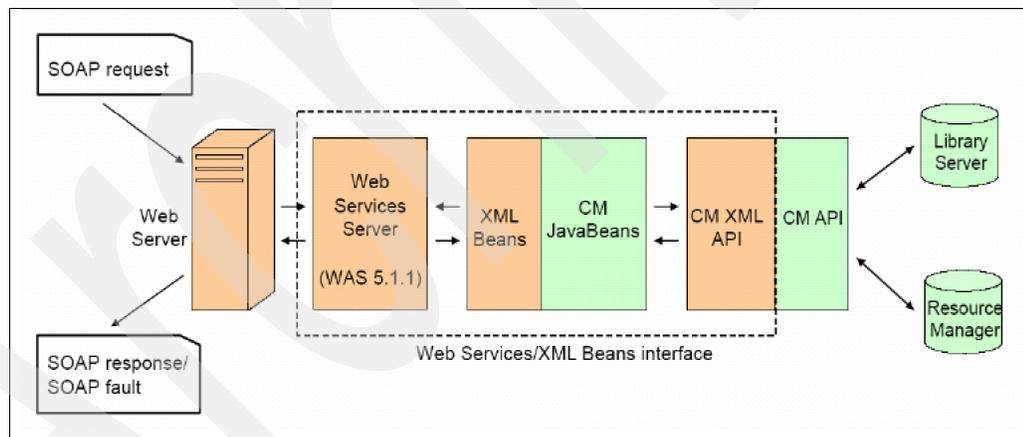


Figure 3-5 DB2 Content Manager Web service architecture

Currently it is implemented as a plug-in to WebSphere Application Server (servlet-based) and supports all core content and document routing functions, such as create, read, update, delete a document or folder, and document routing functions.

There are several tools delivered with DB2 Content Manager to support XML and Web services, as shown in Figure 3-6. The XML schema mapping tool simplifies the process of defining the DB2 Content Manager storage schema (itemtypes) to support incoming XML documents. It dynamically maps an existing XML schema to a DB2 Content Manager itemtype through the use of the graphical utility.

After the mapping is generated, XML documents adhering to the mapped XML schema can be captured, shredded, stored, and managed in DB2 Content Manager automatically, with no human interaction via the API. The schema mapping utility generates XSLT scripts and provides schema import, export, and XSLT transformation services.

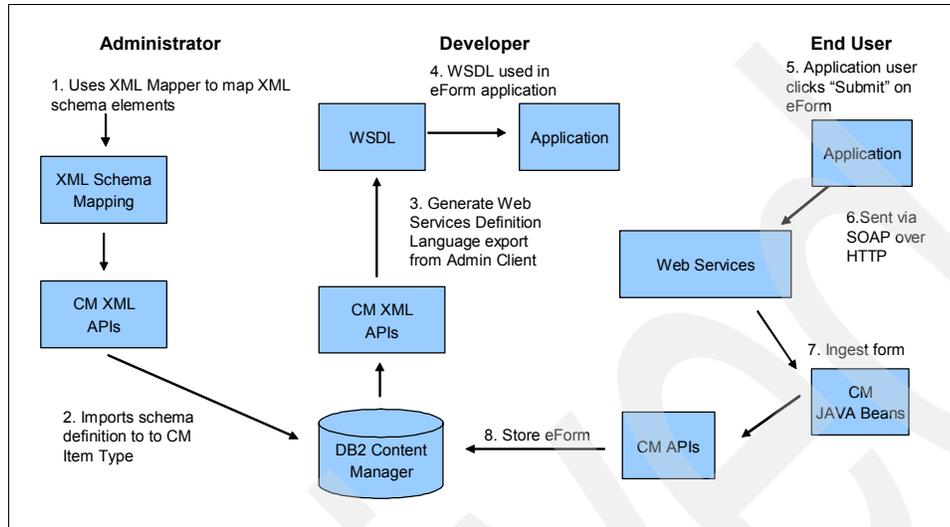


Figure 3-6 XML tools for DB2 Content Manager

After the storage schema is defined, the definition can be exported for use with applications. Administrators can generate the Web Services Definition Language (WSDL) from a DB2 Content Manager itemtype definition. This can be used by developers in their applications.

DB2 Content Manager provides an open, published, consistent object-oriented set of APIs for application integration. This makes it possible to connect and integrate with several business applications such as Adobe Forms, PeopleSoft® EnterpriseOne, SAP Netweaver, CRM Siebel, Lotus Domino, and MS Exchange.

3.2.2 IBM DB2 Content Manager OnDemand

DB2 Content Manager OnDemand is an automated archival and retrieval system that is used to store printed output such as reports, statements, invoices, and image documents. Content Manager OnDemand is optimized to capture, search, present, and manage large collections of small objects. Therefore, it is primarily an archive for *computer output*.

The core server and client components offer the following powerful capabilities:

- ▶ Report and document capture handles multiple data types, and is easy to configure using graphically defined templates.
- ▶ Search, view, print, and fax options are varied and easy to use, including support for annotations and logical viewing.
- ▶ Storage management is automated, optimized for cost and retrieval benefits, and provides immediate compression results.
- ▶ An administrative client provides central control of servers, configuration, report definition, and security management.

Figure 3-7 shows the process of capturing document or reports from a pool.

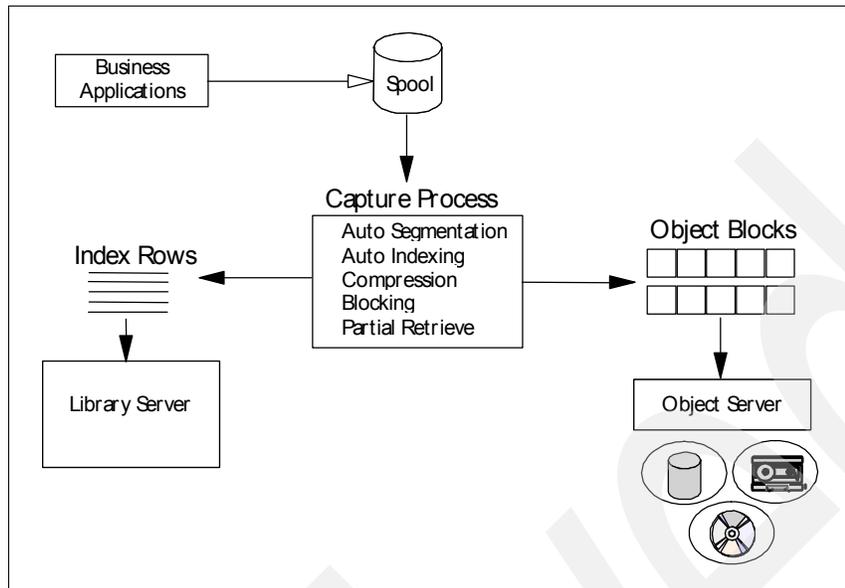


Figure 3-7 Capturing process in OnDemand

Administrators and users can have access to the data stored with either the OnDemand Web client (OnDemand Web enablement kit) or OnDemand Windows client.

Advanced functions of OnDemand include report management and distribution, CD-ROM production, PDF indexing, and integration with Xenos transforms. Report Distribution is a feature that provides an easy way to automatically group reports and portions of related reports together, organize them, convert the report data into different formats, and send them through e-mail to multiple users or make them available for printing.

OnDemand supports DB2 UDB, Oracle, and SQL Server as the index database and AIX, Sun™ Solaris, HP-UX, Red Hat Enterprise Linux, SuSE Linux Enterprise Server, and Windows as the platform. It also supports z/OS with DB2 as the database.

However, OnDemand does not provide:

- ▶ Document management features, such as editing and versioning of documents
- ▶ Case or folder management
- ▶ Integrated workflow
- ▶ A set of open APIs for external programming

These functions are covered by Content Manager.

3.3 Document management

Documents are at the heart of many complex and critical business operations, including product development, financial reporting, marketing, customer and channel support, facilities management, and regulatory compliance. Document management services are key to the success of these business operations and include version and rendition management, auditing, compound document support, and life-cycle management based on your organization's rules for document management.

This section describes the core document management products of IBM. For other products of IBM related to this topic and complementary solutions, see the following Web page:

<http://www-306.ibm.com/software/info/contentmanagement/business/Documentmanagement/Documentmanagement/index.html>

3.3.1 IBM DB2 Document Manager

IBM DB2 Document Manager provides an enterprise wide document management platform to manage business documents. It includes features such as check-in and check-out, versioning and revisioning, audit trails, and compound documents, as well as lifecycle and rendition management. DB2 Document Manager and its extended document services are based on DB2 Content Manager as illustrated in Figure 3-8.

The Document Manager Web-based client is designed to be similar to Windows Explorer. Desktop deployment is automatic, as are software updates, simplifying life for the IT department. The Document Manager interface can be customized to every user, with different sets of features being displayed based on the name and role. Changes to user settings take effect at the next log in. Document Manager integrates with a number of desktop applications (including Microsoft Office, engineering CAD tools, Lotus Notes®, and Microsoft Outlook®). Therefore, users can also interact with DB2 Document Manager directly from common desktop applications such as Microsoft Word.

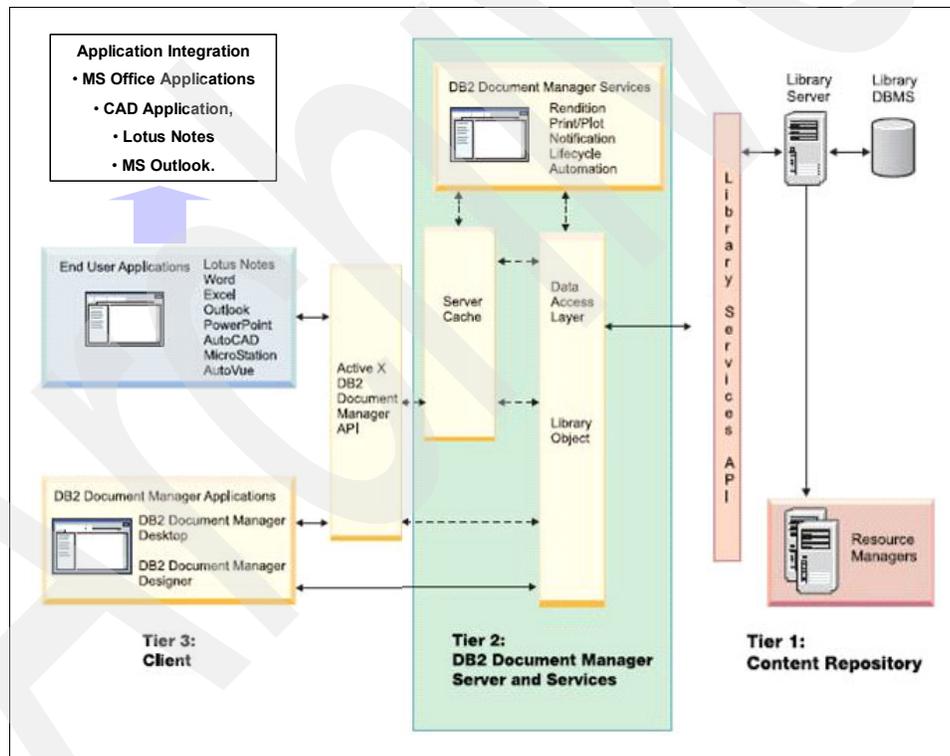


Figure 3-8 DB2 Document Manager architecture

DB2 Document Manager manages the various states and transitions that occur throughout the life of a document, from creation to final disposition. In doing so, DB2 Document Manager supports collaborative document creation and management across groups of globally dispersed creators, contributors, reviewers, and approvers, ensuring that accurate and up-to-date documents are available on demand, where and when they are required.

Many documents are actually composed of multiple interrelated components. For example, a standard operating procedure manual might consist of a word processing file that details procedures, an embedded spreadsheet that defines the scheduling. In order to effectively manage these compound documents, DB2 Document Manager manages the relationships between all of these components, including their individual versions.

The process of document creation itself can vary based on the type of document, which would determine who is involved in the review and approval process. Rules established by various regulatory agencies or other internal or external authorities can also affect this process. These rules-based document creation processes are covered by a concept known as document lifecycle management. DB2 Document Manager supports document lifecycle management by allowing the administrator to configure document lifecycle rules based on a simple, menu-based interface. No programming is required to customize a document approval and release process. These processes can be graphically designed, viewed, and edited.

DB2 Document Manager also provides revision management that ensures that only the approved revision is available to general users. And when implemented along with IBM DB2 Records Manager, when users add documents to the document library and assign them to a document type, the appropriate retention rules and actions are applied. Document retention becomes completely transparent to the user. With Records Manager, users have the added security of knowing that appropriate retention rules are applied.

All activities on a document, such as revisions, reviews, or accesses, are tracked and available for audit. Document security can be changed, based on the new state, to control who can edit, view, or change a document's state. A change in document state can also change folder membership, renditions generated, and notification events triggered. One of the ways this feature could be used is to prevent post-approval changes by an author. Document Manager can generate a PDF file as the only generally accessible version of the approved document.

Instant messaging and presence awareness as a part of document management can be enabled with the integration of the Lotus Sametime® product.

Document Manager also includes a set of services that help in building sophisticated processes. Printing and plotting managers permit users to direct output to special devices or remote locations, whether or not the remote locations have the base application. A rendition manager can transform a document from its authoring format into a format more appropriate for a particular distribution.

For example, a finalized press release might be converted to PDF to e-mail to newspapers, and also be converted to HTML for posting at the corporate Web site. A notification manager issues messages via e-mail or an integrated message window based on Lotus Sametime when specified events occur. Messages can contain document property information, document copies, or links. Sending them saves the e-mail system from the burden of transporting large files and preserves security because the receiver must have access to the document in DB2 Content Manager in order to view it.

3.3.2 Lotus Domino Document Manager

Lotus Domino Document Manager is based on Lotus Domino technology as illustrated in Figure 3-9. There are several document management services such as full text search, routing, access control, directory services, and calendaring (and more) which are released already with the Lotus Domino architecture. Therefore, for a customer who has an established Domino environment already and is looking for a department-wide, Domino-based document management product, this might be an interesting choice.

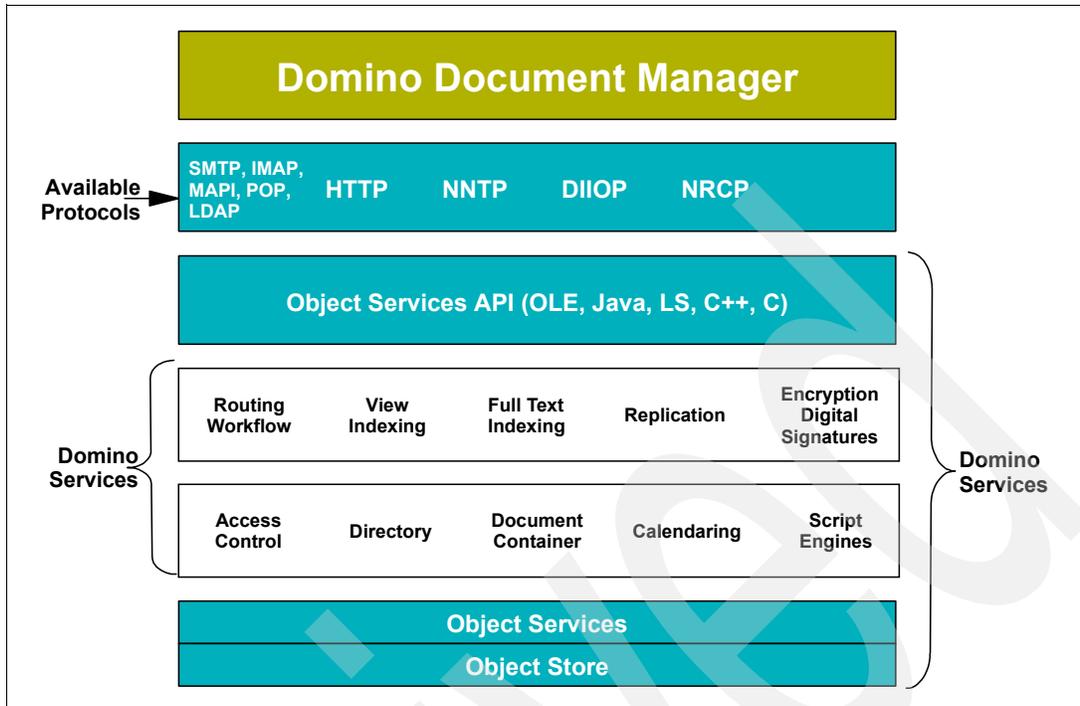


Figure 3-9 Architecture of Lotus Domino Document Manager

It is possible to organize documents for shared access by work teams, to manage versions so that each team member has the latest, and to automate document-driven processes such as review and approval, assembly and publishing, and archiving. The integration with Lotus workflow helps in implementing such document-driven processes. The focus for Lotus Domino Document Manager is to integrate managed documents into the collaboration process by such features as linking documents to tasks, projects, and discussion threads.

Collaboration within and across workgroups is facilitated with several document library services that help teams manage and track documents throughout the lifecycle: collaborative authoring, document check-in and check-out, version control, revision history, audit trail creation, publishing and archiving capabilities, threaded discussions, and offline support.

Lotus Domino Document Manager can be integrated with Lotus Notes and MS Office. This makes it even easier to implement Lotus Domino Document Manager for people working already with Lotus Notes. Actions such as “save as” and “open as” from within users’ familiar desktop applications now tie directly to Lotus Domino Document Manager check-in and check-out features.

Instant messaging and presence awareness as a part of document management can be enabled with the integration of the Lotus Sametime product.

Lotus Domino Document Manager can use Domino as the repository for customers with highly collaborative environments, or it can be integrated with DB2 Content Manager in installations where a more comprehensive repository is required.

Lotus Domino Document Manager open application programming interfaces (APIs) allow you to use LotusScript, Microsoft Visual Basic®, Visual C++® and certain other programming languages for customization.

3.4 IBM DB2 CommonStore

IBM DB2 Content Manager CommonStore helps to seamlessly integrate SAP, Lotus Domino, and Exchange Server with IBM archives. CommonStore integrates with the target system to off-load data on to an external storage. This improves the performance of the target system and cuts down storage costs.

There are three independent modules available for CommonStore:

- ▶ CommonStore for Exchange Server
- ▶ CommonStore for Lotus Domino
- ▶ CommonStore for SAP

CommonStore is a middle ware server between SAP, Lotus Domino, Exchange Server, and the back-end archive management system. CommonStore does not store data or document, but defines and manages what to archive, when to archive, and how to archive from the mail system to the back-end archive management system.

Three back-end archives are supported:

- ▶ IBM Tivoli Storage Manager
- ▶ DB2 Content Manager
- ▶ DB2 Content Manager OnDemand

Figure 3-10 shows the modules of the CommonStore server.

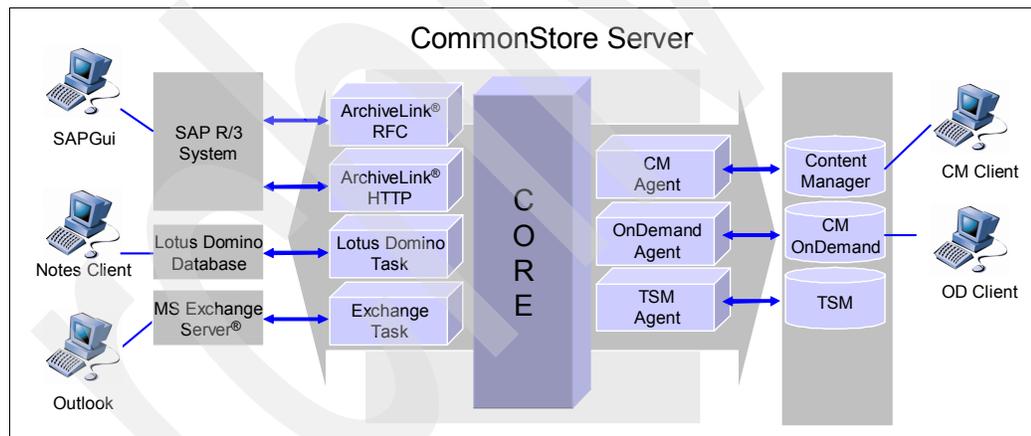


Figure 3-10 Basic architecture of the CommonStore server

3.4.1 CommonStore for Exchange and CommonStore for Lotus Domino

CommonStore for Exchange Server and for CommonStore for Lotus Domino helps with e-mail archival and retrieval. It manages e-mail server growth by automating e-mail archival, thus trimming down the size of online e-mail storage. Archival can be configured so that:

- ▶ The entire mail document is archived, including attachments (it leaves the mail header in the mail system as a placeholder).
- ▶ Only the attachments (the mail body) remain in the mail system).

The archival can be initiated by individual users on any message or document they select, or it can be driven by automated pre-scheduled policies without user involvement. The single instance store feature assures that messages are archived just once. Attribute mappings allow saving certain message properties in the content repository, such as “subject”, “sender”, and “receiver”.

There are several options for deletion that are available during the archival process:

- ▶ **Attachment:** URL links will be inserted for attachments. It is also possible to archive the entire mail and to remove the attachments only.
- ▶ **Body:** This option is only available when archiving entire messages. A URL allows viewing of the message without restore. An abstract of the body can be created in the stub.
- ▶ **Entire message:** Complete messages will be deleted from the server. No reference will be maintained in the mail system. Only the search functionality in the native repository can give back the access to the message.
- ▶ **Nothing:** Messages remain unchanged.
- ▶ **Intelligent abstracting:** This is another archiving option. A short summary of the mail is inserted in the mail body. The intelligent text analysis is based on IBM's Text Analysis Framework (TAF), which identifies most relevant sentences.

Direct access to archived e-mails using a Web browser or mail client is provided as well.

With DB2 CommonStore, Version 8.3, IBM delivered a new integration with DB2 Records Manager, enabling you to declare e-mail messages and attachments as records while archiving them, either with user-based selection, drag-and-drop activity, or fully automated without user involvement. With this built-in integration, you can manage, retain, and dispose of e-mail as records based on regulatory, legal, and corporate requirements, improving operational efficiency while addressing compliance requirements. More details about this area are given in the IBM Redbook, *E-mail Archiving and Records Management Integrated Solution Guide Using IBM DB2 CommonStore and DB2 Records Manager*, SG24-6795.

There is a technical comparison of CommonStore for Lotus Domino and CommonStore for Exchange available with the following link:

<http://www-128.ibm.com/developerworks/db2/library/techarticle/dm-0602tham/#mapping>

3.4.2 CommonStore for SAP

CommonStore for SAP is a middleware server between the SAP ArchiveLink™ interface and a back-end archive.

As your SAP database grows, so does the volume of SAP related data. CommonStore for SAP relieves the pressure on your SAP system's database and improves its performance by offloading inactive data to an external back-end archive. In addition to data archiving, DB2 CommonStore can manage a wide range of information, including:

- ▶ Inbound documents such as supplier invoices
- ▶ Outbound documents normally printed and sent to their respective recipient
- ▶ Reports and print lists such as journals
- ▶ Desktop files created by PC applications such as MS Office, and also other documents created outside of the SAP system

With the help of SAP Document Finder it is possible to search from the SAP GUI for all enterprise content stored in Content Manager or OnDemand, not just archived documents from SAP.

DB2 CommonStore for SAP, Version 8.3, is certified by SAP AG for current SAP releases, including SAP R/3®, SAP R/3 Enterprise™, mySAP™.com, and the SAP NetWeaver® framework. It supports all SAP system platforms and manages all types of data and documents defined in the SAP ArchiveLink.

3.5 IBM DB2 Records Manager

DB2 Records Manager brings formal, structured records retention and disposition for both electronic and physical information assets. When used within a context of clear and consistent corporate policy, it can reduce litigation risk and evidence discovery costs and help you demonstrate compliance with government and industry regulations.

Records Manager provides one central location for record classification and retention policies. It is a product for organizations that must demonstrate compliance with design criteria such as the US Government standard DoD 5015.2, to improve management controls over both electronic and physical records and to apply retention and disposition management to electronic and physical information.

Records Manager lets you:

- ▶ Declare and classify records from using fully automatic procedures to manual processing.
- ▶ Apply retention periods and disposition to electronic and physical information
- ▶ Apply records management consistently in a manner that is non-intrusive to current business practices and IT environments.
- ▶ Deliver a single e-records solution integrated across multiple applications, including IBM and non-IBM systems as well as Windows and non-Windows applications to apply the complete lifecycle management to information assets.
- ▶ Apply complete lifecycle management to information assets.

Figure 3-11 shows an example of manual declaration and classification of records.

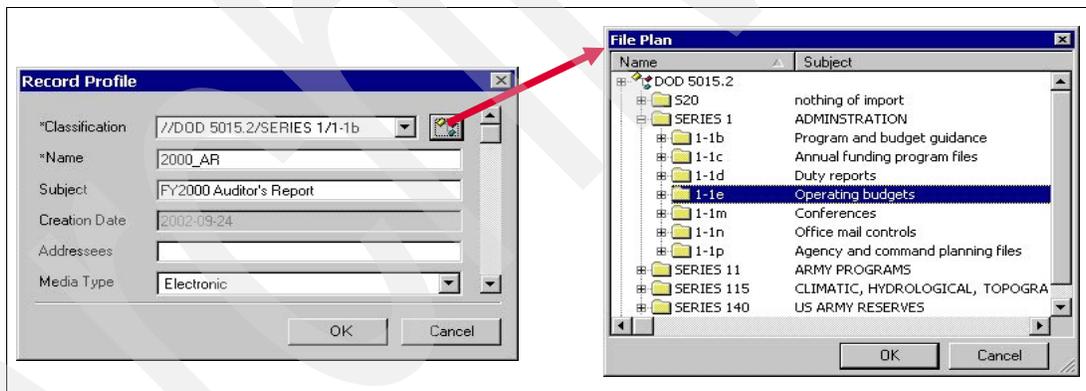


Figure 3-11 Declaration and classification of records

The engine approach with e-records enables applications without installing a desktop records management application. The Recordkeeping Methods Modeling (RMM) inside of Records Manager includes easy adoption across business models or geographic specific methods, unlike traditional records management applications (RMAs), which generally have a fixed model and must be modified for unique recordkeeping methods. Also, unlike most RMAs, the Records Manager engine does not store or extract records from the host business application. It applies retention and disposition rules to the documents within the host business application's repository, ensuring that the security of the document is not disturbed.

It is integrated with applications such as IBM DB2 Content Manager, IBM DB2 Document Manager, IBM DB2 CommonStore for Lotus Domino, and IBM DB2 CommonStore for Exchange Server. Records Manager APIs facilitate the integration with any application that requires its record keeping capabilities.

Federated Records Management

Records management has clearly come to the forefront of corporate priority as a result of sweeping legislation and rigorous regulation around business practices and processes. But the reality is that the information that should be managed as a record is typically stored in multiple disparate systems.

IBM Federated Records Management delivers a multi-repository solution to help organizations centrally manage records distributed across multiple, disparate content management repositories and business applications. Records-enabled business applications leave records in their native repository and keep business processes intact, preserve vital security and unburden the user from records management overhead. IBM Federated Records Management features include the ability to manually declare records, apply holds to suspend disposition processing, and perform searches to locate content or records.

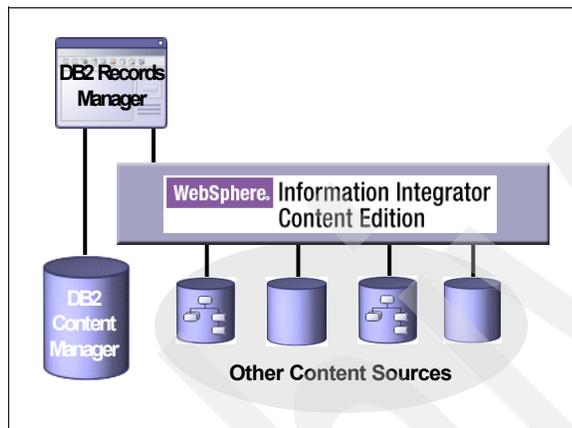


Figure 3-12 Basic architecture of Federated Records Management

Federated Records Management is a combination of DB2 Records Manager, DB2 Content Manager, WebSphere II Content Edition and Services Assets that tie the products together as shown in Figure 3-12. A base Federated Records Management system consists of only these components, plus the connectors for the customer's content sources such as IBM Content Manager, Documentum, Filenet, Opentext, or Hummingbird.

3.6 IBM Workplace Web Content Management

Workplace Web Content Management™ is a Web-based, multi-user authoring tool used to create, control, and publish content to Web sites. By using Workplace Web Content Management, the development and delivery of information is accelerated, consequently allowing users to drive down the cost of creating and managing their Web site contents, which usually exist in different forms and formats within the companies. With Workplace Web Content Management, the information can be freely distributed and instantly updated across all existing applications, including Internet, intranet, and extranet Web sites.

Workplace Web Content Management separates the design and presentation from content creation. This allows the creation of content once, and the display of the same thing with a different look and feel. Business users can create and maintain their Web sites easily, without worrying about what the impact of their contents in the Web sites looks and feels like. This also guarantees a consistent presentation because the contents remain unchanged even when the design changes.

Lifecycles can be created containing draft, published, and archived Web content objects. A typical Web content workflow contains a draft, publish, and archive life stage.

IBM Workplace Web Content Management lets you manage and render content within a WebSphere Portal environment. Content within a traditional Web-based delivery environment consists of linked HTML. In Workplace Web Content Management, the content is broken into reusable objects and linked components. At a high level, content within Workplace Web Content Management is considered as the following:

- ▶ Sites and site areas (site framework components)
- ▶ Presentation and authoring templates (site framework components)
- ▶ Workflow and workflow stages (content management components)
- ▶ Components (content resource components such as files, menus, and templates)
- ▶ Content (the combination of text, components, and resources)

To manage and store the associated framework, content objects, and resource definitions, Workplace Web Content Management uses a dedicated repository, which is maintained and accessed through a database management system (DBMS). Figure 3-13 shows the lifecycle of Web content and the integration with DB2 Content Manager and IBM Storage Management.

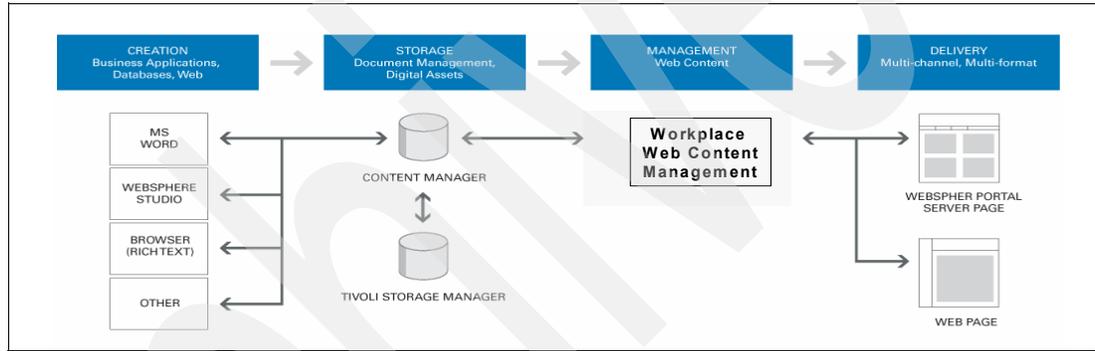


Figure 3-13 Lifecycle of Web content

It enables you to capture, manage, and reuse all forms of content across diverse applications, business processes, and platforms to deliver integrated, consistent, and on demand information to customers, partners, and employees. Any Content Manager content can be easily published on the Web. Some advantages of using Content Manager as the repository for Workplace Web Content Management are:

- ▶ Saving time and money by reusing content instead of recreating it
- ▶ Integrated hierarchical storage management and archiving, such as using Tivoli Storage Manager
- ▶ Integrated retention and disposition management for the content

3.7 IBM Workplace Forms

Forms are vital components of many business processes. Forms provide the interface for providing crucial information, such as requests, approvals, who, what, how many, when, and so on. Forms are significant factors in determining how efficiently a process runs and in turn, how smoothly your entire business operates.

With Workplace Forms™, you can create, deploy, and manage XML forms-based processes. You can design standards-based, secure forms, by using an easy-to-use WYSIWYG form designer. Workplace Forms is 100 percent XML, and supports JSR-168, JSR-170, Java™ 2 Platform, Enterprise Edition (J2EE™), and Web services.

Workplace Forms stores form documents in a class of XML documents called Extensible Forms Description Language (XFDL). XFDL was defined to standardize the process of digitally representing complex forms, such as business and government forms. XFDL supports high-precision layout, integrated computations and input validation, digital signatures, and other features.

On the client side, Workplace Forms applications can be viewed through a standard Web browser, or through a browser with the Workplace Forms Viewer plug-in for a richer user experience. When using Workplace Forms Viewer, you can work with forms offline. Digital signatures are also supported. Data, logic, and presentation can be signed, allowing for a complete transaction record.

Workplace Forms provides a single envelope for all XML components (presentation, business logic, data, and XML attachments). You can build dynamic e-forms that can branch or change course, depending on user input. Security features help ensure your transactions are safe and have not been tampered with. You can store and archive entire e-form records, and parse data for later reuse. A built-in compression feature helps reduce form size.

Workplace Forms consists of three modules:

- ▶ **IBM Workplace Forms Designer:** This is the primary tool for creating e-forms. Workplace Forms Designer provides an easy-to-use interface, with user-friendly features, such as drag-and-drop creation of form components.
- ▶ **IBM Workplace Forms Server:** This consists of three components:
 - The Workplace Forms Server: API provides integration capabilities.
 - The Workplace Forms Server: Deployment Server is a light-weight installation system for deploying the IBM Workplace Forms Viewer to the user's desktop.
 - The Workplace Forms Server: Webform Server is a server-side component that translates XFDL into HTML/JavaScript, providing a zero-footprint “thin client” rendering system.
- ▶ **IBM Workplace Forms Viewer:** This is a browser plug-in that provides enhanced features to users working with Workplace Forms applications.

Tip: It might help to understand how Workplace Forms Designer relates to the similarly named IBM Workplace Designer, which IBM introduced in 2005. Both Workplace Forms Designer and Workplace Designer are development tools, designed to build graphical, user-facing IBM Workplace applications. But there are significant differences between the two. Workplace Forms Designer lets you create XML e-forms for automating forms-based business processes. Workplace Designer is intended for script developers who want to build re-usable components (deployed as portlets) for IBM Workplace products, such as Workplace Collaboration Services and Workplace Services Express.

Workplace Forms consists of document-centric component technology, designed to be integrated with middle ware, such as portals, content repositories, such as IBM Content Manager, and workflow systems. Figure 3-14 illustrates how this integration looks like with the Content Manager Content Repository.

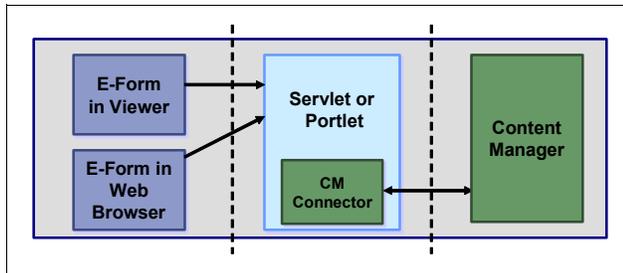


Figure 3-14 Forms integration with DB2 Content Manager

3.8 Enterprise Search and Content Discovery

Enterprise search is about finding the most relevant information from the plethora of enterprise information stored in file systems, content repositories, databases, collaboration systems, applications, and the company intranet. Finding the right information can be frustrating, time consuming, and costly. Customer satisfaction decreases as average call time to the call center or client wait time increases. Or worse, business opportunities are missed when a bad decision is made in the absence of all the relevant information. Employee productivity erodes when they spend too much time searching information as opposed to solving business problems.

Enterprise search is different from Internet search. Searching enterprise sources means developing different techniques to determine document relevancy and taking into account different security models and the many different data sources and file types. Even the most successful Internet search techniques such as page ranking are not optimized for an enterprise environment where documents are not generally as interlinked to each other. To address the enterprise environment, IBM has developed new information relevancy techniques to deliver high quality results to users searching for meaningful information in their company's vast array of enterprise content.

3.8.1 IBM WebSphere Information Integrator Content Edition

WebSphere Information Integrator Content Edition, which is part of the WebSphere Information Integrator portfolio, has the capabilities to provide enterprise applications with relevant content, such as documents, images, audio, video, and other unstructured and semi-structured information stored in multiple, disparate repositories throughout the enterprise.

WebSphere II Content Edition provides a single, Java-based, bidirectional interface to access many different content repositories (such as IBM Content Manager) and workflow systems, making it easy for application developers to integrate those sources into new or existing enterprise applications. The product includes prebuilt Web components, making it even easier to include WebSphere II Content Edition capabilities into Web applications, including the ability to read and update content. Other capabilities include:

- ▶ Cross-repository federated searching
- ▶ Virtual repositories to work with content from multiple repositories
- ▶ Cross-repository event services
- ▶ Data dictionary for mapping metadata fields across repositories
- ▶ XML import and export into a repository neutral format
- ▶ Automatic content conversion to browser-ready formats

As shown in Figure 3-15, WebSphere II Content Edition's services oriented architecture can be described in terms of core integration services underlying a rich set of multi-repository federation services with access to the system via developer and user services, all while maintaining strict security for the content being integrated.

Integration services provide a single, consistent interface to the underlying content repositories, including content, functionality, and workflow capabilities. Integration services expose a super-set of content management and workflow functionality and also maintain the awareness of both the available repositories and the functional capabilities of each repository.

This means that your client applications are not limited to a least common denominator of repository capabilities but can discover the capabilities available for any particular repository item. By defining a complete, uniform model through which this functionality can be accessed, applications leveraging WebSphere II Content Edition can readily expose the full capabilities of existing repositories, regardless of the underlying repository or vendor. Furthermore, applications built on WebSphere II Content Edition are "future-proofed" against changes to the enterprise infrastructure such as upgrades to back-end systems, migration from one system to another, or acquisition of new systems.

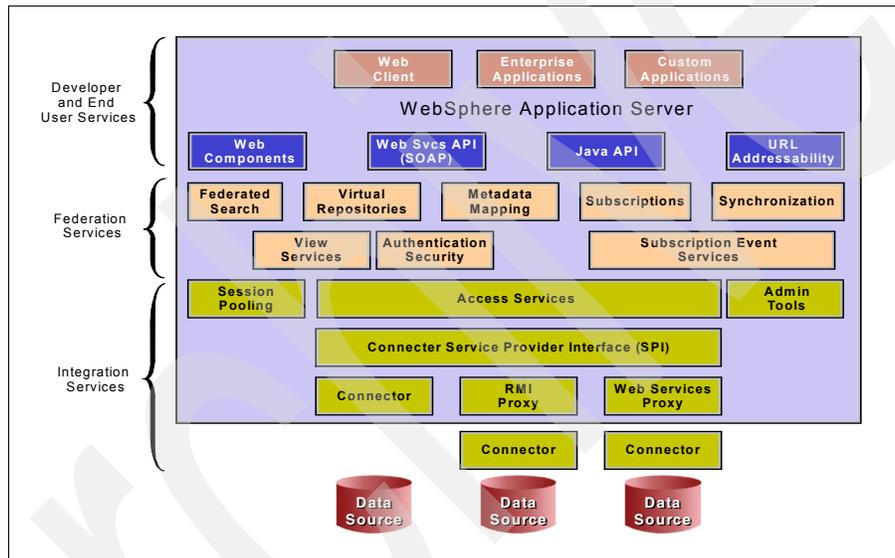


Figure 3-15 Modules of WebSphere Information Integrator Content Edition

The following operations are available:

- ▶ Search for content: Perform parametric and full-text searches against one or multiple content repositories.
- ▶ Capture content: Add content and metadata to repositories.
- ▶ Control content: Perform library functions such as check-in or check-out and copy or transfer folders and documents within a repository or across repositories while maintaining properties, versioning information, and other content attributes.
- ▶ Retrieve content: Retrieve content and associated meta-data values from repositories in the content's native format or in an XML document.
- ▶ Update content: Make changes to content and update meta-data values, annotations, and security settings while maintaining version control.
- ▶ Manage content hierarchies: Create and delete folders, file and un-file content in folders, retrieve folder contents, and update folder properties.

- ▶ Search for work items: Perform parametric searches against one workflow engine or federated searches against multiple workflow engines.
- ▶ Create new work items: Initiate new instances of workflow processes and apply meta-data values and content attachments.
- ▶ Retrieve work items: Retrieve work items and any attached content from an inbox or specific queues or steps in the workflow process.
- ▶ Update work items: Make changes to work items including meta-data and attachments. Perform actions on the work item such as locks, suspend/resume and dispatching.
- ▶ Audit: All actions initiated through WebSphere II Content Edition can be audited at various different levels with all the pertinent information such as the time, the user, the specific action taken and item being accessed.
- ▶ Maintain security: Ensure that users access only authorized content and work items by taking advantage of the security features inherent in the underlying system.
- ▶ Manage sessions: Log on and log off to content repositories and workflow systems with password encryption over the wire. Handles session pooling.

It is important to understand that WebSphere II Content Edition itself provides access to these capabilities and does not provide the implementation. That capability is provided rather by the back-end repository.

The main module of integration services is an architectural hub called Access Services shown in Figure 3-16. Access Services is implemented as a stateful session EJB™ with one instance per session. The J2EE application server provides EJB clustering to support load balancing and high availability, and distributed network communications to support various network topologies and geographic scenarios. An Access Services instance defines a single WebSphere II Content Edition session and brokers access to disparate enterprise repositories by relaying application requests to the appropriate repository via connectors. Access Services aggregate the results of multi-repository application requests and return this information to the client application, along with any requested metadata and content in the desired format.

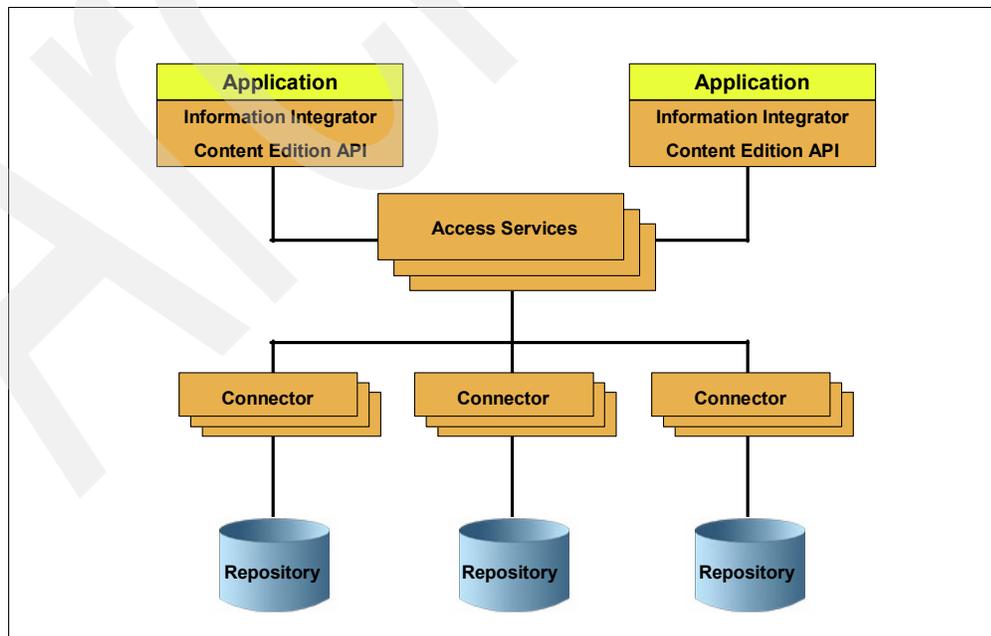


Figure 3-16 Access Services layer as part of integration services

Access Services also serves as a configuration hub, communicating with a configuration server to determine the active configuration of the system. This allows the configuration data to remain in a centralized, fail-safe service while being propagated out to the other services as required.

WebSphere II Content Edition must translate the requests made to Access Services (such as searching or capturing content) to the vendor-specific APIs of content repositories and workflow engines. This translation is done by connectors, which also normalize the results of those operations and return the data to Access Services. WebSphere II Content Edition includes connectors for a wide variety of popular content repositories and workflow engines. They are also extensible to support unique or nonstandard implementations. If you want to develop a new connector, there is a connector SDK to help you do that.

Connectors are available in the product to the following repositories:

- ▶ IBM DB2 Content Manager and Content Manager OnDemand
- ▶ IBM DB2 WebSphere MQ Workflow
- ▶ IBM Lotus Domino and Domino Document Manager
- ▶ FileNet
- ▶ EMC Documentum
- ▶ Microsoft Index Server/NTFS and Sharepoint Portal Server
- ▶ Open Text LiveLink
- ▶ Stellent Content Server
- ▶ Interwoven TeamSite
- ▶ Hummingbird Enterprise DM

The federation service, which is built on the integration services, make it easier to deal with multiple sources of content and workflow automation at the same time. Federation services include:

- ▶ Federated search for performing a single for all relevant content across many repositories
- ▶ Data maps, which translate between the disparate indexing schemes of each repository
- ▶ View services for on-the-fly rendering of content
- ▶ Virtual repositories for virtually reorganizing content to support new business initiatives
- ▶ Subscription event services for providing event notification of changes in the repositories

The developer and user services deliver the capabilities of WebSphere Information Integrator to the applications that require them. These services include an out-of-the-box Web client, Web components for quickly building custom Web applications, and APIs.

3.8.2 IBM WebSphere Information Integrator OmniFind Edition

Information is isolated in multiple content sources typically created by individual departments, but the requirements of information consumers typically cut across an organization. Also, the vast majority of this information is unstructured (not indexed). Another challenge is that conventional search and browse experience is not good enough.

WebSphere Information Integrator OmniFind™ Edition provides the capabilities for searching multiple, especially unstructured (and also structured) data sources with a single query from the Web browser. It returns a consolidated, ranked result set for quick and easy location of the information that is required. WebSphere Information Integrator OmniFind Edition components collect information from throughout the enterprise and make it available for searching.

It does this by extracting the documents from their original source, parsing and analyzing the content, then building a collection (index) that is optimized for speed and search quality. By entering a query in a Web browser, a user can simultaneously search local and remote databases, collaboration systems, content management systems, file systems, and internal and external Web sites. The resulting set of document links can be used to retrieve the original document from its native repository. WebSphere Information Integrator OmniFind Edition also addresses the requirement for stringent security safeguards to protect content from unauthorized access using.

Figure 3-17 summarizes the phases and key technologies used to prepare the enterprise content for search. The content is first extracted from its source through a process called “crawling,” similar in concept to the crawlers used for the Web but also applied to non-Web data sources. The content is then parsed and tokenized to identify individual words. Next the documents are optionally categorized.

The documents are then further annotated with features found in the text. This is where the advanced text analytics are applied. A document might be annotated to identify proper nouns, dates, relationships between words, and so on. After the documents have been tokenized and annotated, they are ready for indexing. Global analysis is performed on the entire set of documents to determine its static ranking. A common task would be to perform link analysis on Web documents for example. The more documents that are linked to it for a particular reference raises its rank for that reference (or keyword). And lastly, the index is made available for searching.

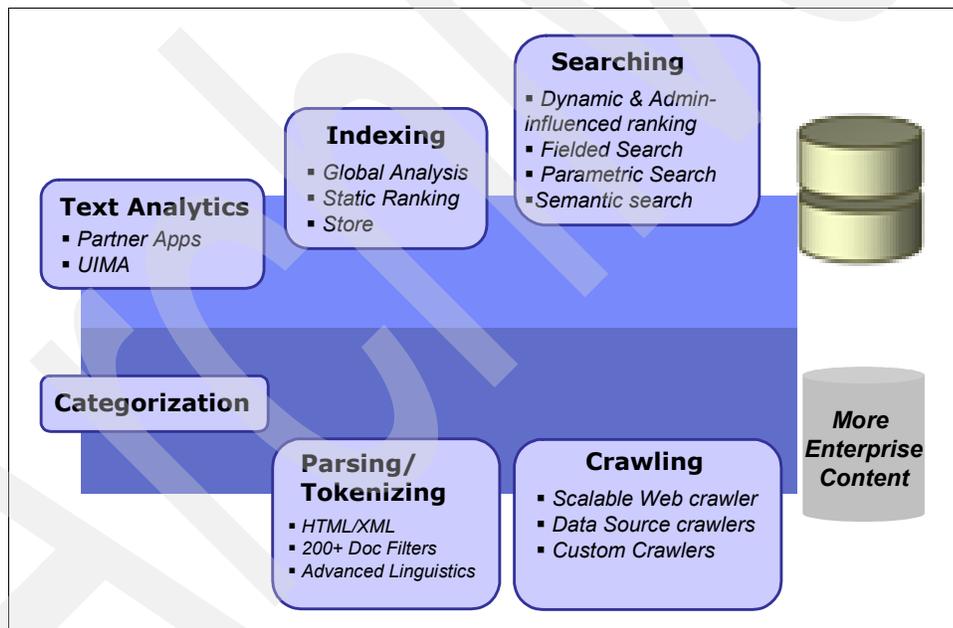


Figure 3-17 Basic concepts of WebSphere Information Integrator OmniFind

WebSphere Information Integrator OmniFind and its search application, delivered with it, provides a user interface that can exploit and present the set of capabilities provided by parser and tokenizer, such as stemming of verbs, lexical affinities (synonyms), stop-word elimination, dynamic summary, relevance ranking, security checking, or quick links within the search result list. This includes functions for determining the meaning or relevance of words, character normalization, such as normalizing capitalization, and German umlauts as well.

There is also a plug-in for Google search available to extend Google desktop search capabilities to an full Enterprise Search powered by the technology of WebSphere Information Integrator OmniFind Edition.

The range of data sources that are supported by WebSphere Information Integrator OmniFind Edition include file systems, content repositories, databases, collaboration systems, intranets, extranets, and public-facing corporate Web sites, including:

- ▶ Web (HTTP/HTTPS)
- ▶ News groups (NNTP)
- ▶ File systems
- ▶ Domino databases
- ▶ Microsoft Exchange public folders
- ▶ DB2 Content Manager
- ▶ EMC Documentum, FileNet CS and P8 CM, Hummingbird and OpenText Livelink Enterprise Server
- ▶ Various databases such as IBM DB2 UDB, Informix® Dynamic Server, Oracle Database Server, Microsoft SQL Server, Sybase, and Software AG Adabas
- ▶ WebSphere Portal 5.1 Web pages and WebSphere Portal 5.1.0.1 Document Manager
- ▶ Workplace Web Content Management
- ▶ Lotus Notes/Domino Server, Lotus Domino Document Manager and Lotus Domino QuickPlace®
- ▶ IMS 7.1

Note: New sources are continually being added, and readers should refer to the following Web site for an up-to-date list of supported data sources:

http://www.ibm.com/software/data/integration/db2ii/requirements_womnifind2.html

3.8.3 IBM WebSphere Content Discovery Server

The basic concept of Content Discovery Server is to reduce search time and to increase customer satisfaction. See Figure 3-18.

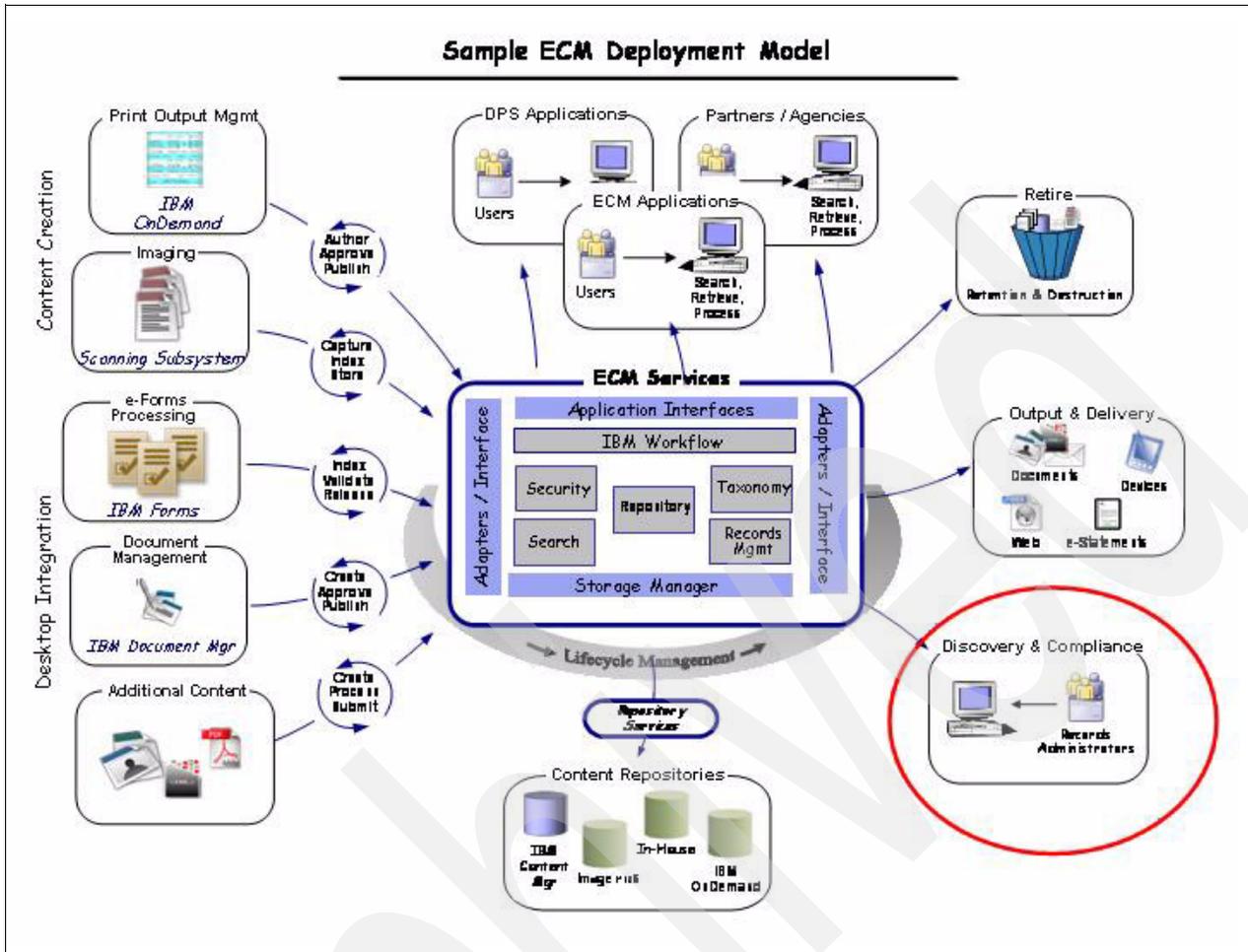


Figure 3-18 Discovery

Content Discovery Server offers an intuitive way to discover information, by using natural language and efficient correlation mechanisms. Contextual understanding interprets query intent and application context to help people finding information based on what they mean as opposed to what they say. Adaptive presentation guides the discovery process by presenting answers, navigation options, and proactive guidance in a format that helps people take action. It allows experts to monitor the online experience and make real-time improvements without reliance on IT. A prepackaged line of business modules includes industry vocabularies, configuration logic, and application user interfaces, as follows:

- ▶ Content Discovery for Commerce helps online retail and catalog companies convert shoppers into buyers. It taps product attributes, such as brand, department, and price, and descriptive merchandising text to deliver relevant products regardless of spelling and grammar. It helps shoppers find the products that meet their requirements by dynamically analyzing the underlying catalog to present intuitive browsing options that allow shoppers to iteratively select the attributes that matter most to them (such as brand, price, size, and style) and presenting intuitive hints that guide related shopping searches.

- ▶ Content Discovery for Self Service is an adaptive search engine for customers and employees alike to go through any kind of information gathering process. The customer can find a solution to a boggling problem, and the employee can get an answer to a specific item produced by his company. Because of the adaptiveness of the search engine, it detects when a search requests goes in circles, and offers an escalation process, for example, an e-mail thread. This e-mail then gets “milled” with even more time consuming algorithms, which might come up with a new solution that then gets sent back to the customer or employee.
- ▶ Content Discovery for Online Support enables people to solve problems without engaging a customer service agent and manages escalation when additional help is required. It combines a real-time understanding of user intent and application context to optimize relevance of information that is delivered. Personalization of information can be offered based on profiles (for example, products owned).
- ▶ Content Discovery for Contact Centers delivers contextually relevant information located across an organization to resolve support inquiries without escalation based on a rich understanding of customers’ and agents’ search requests, support e-mails, and case comments. It combines a real-time understanding of user intent and application context to optimize relevance of information that is delivered. Personalization of information can be offered based on roles (customer versus support representative) and profiles (products owned).
- ▶ Content Discovery for Case Resolution intercepts online service requests and responds to people with answers, thus avoiding escalation to the contact center. It is an online Web form solution that provides e-mail auto-response and managed escalation to leading CRM solutions.

These modules can be integrated into Portals as well as into existing CRM and other Call Center applications. WebSphere Content Discovery Server consists of four main services:

- ▶ Classification Server: This module classifies text based on predefined categories and subjects which usually are associated with solutions for problems.
- ▶ Search and Interaction Server: The search server looks after the content on the different systems. The interaction server does parsing, stop word removal, and presentation of results.
- ▶ Management Console: This module is used to prepare and maintain business rules and as well for monitoring.
- ▶ Interactive Analytics: This module customizes reports by drilling down, filtering, and sorting report data. It also generates a wide variety of graphs for each report and can export the reports into MS Excel®, CSV, PDF, and HTML formats. It is possible to create custom reports based on predefined or custom metadata.

Figure 3-19 illustrates the interaction of these services based on a user query entered into the system.

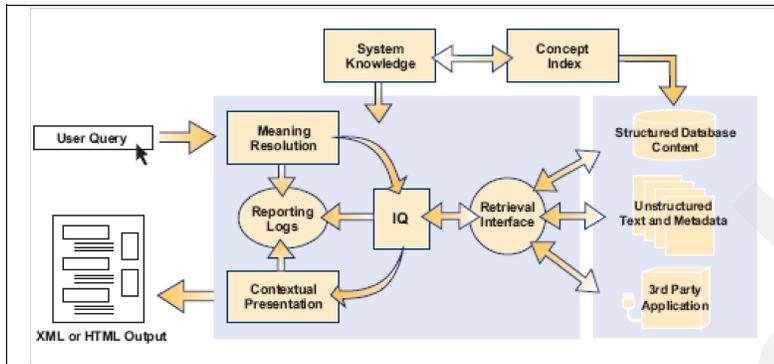


Figure 3-19 interaction and Search Server

A user query entered into the system gets analyzed and parsed (Meaning Resolution). System knowledge, entered by the management console or derived from presearch preparations of the content (indices, keywords, relations, cases, solutions, any kind of possible relationships) is added to the seek formula. The retrieval interface (search server) then processes this request and goes through the content. As one can see here, it is not just structured and unstructured data, but also information from third party applications, that can be retrieved. The returned results go through the interaction server again to be sorted, checked on its relevance, put in context (contextual presentation) and made visible. The results are presented to the user.

3.9 DB2 Content Manager VideoCharger

VideoCharger™ provides real-time multimedia streaming and enhances rich media capabilities of Content Manager. VideoCharger delivers high-quality audio and video streams over corporate intranets or the Internet. It supports multiple media formats including MPEG-1 to MPEG-4 and Apple QuickTime 6. Videos are “pushed” by the server over the network to the client, similar to a broadcast environment where a video stream is started by a play command and will continue until stopped. This contrasts with most file servers today where the data is “pulled” by the client issuing successive “reads” to the server. Therefore, with VideoCharger it does not require that the file be downloaded or saved before being played by the client software. The flexible architecture allows system performance and price performance for high volume video. Streaming video is pushed through “data pumps.” With each data pump, a greater volume of streaming content can be pushed.

The “Filter Architecture” allows pluggable support for new codecs, custom client support, live capture, watermarking, encryption, and support for proprietary codecs and formats. IBM Filter technology “future proofs” investment, to take advantage of new technology. The support of protocol standards provides streaming flexibility for low quality bit rate, mid-band or high quality betrayed video streaming.

A comprehensive Web based administration and configuration facility provides loading, search, and query functions for systems management.

New in V8.3, VideoCharger introduces a streaming technology called Adaptive Rich Media Streaming (ARMS). With ARMS, media is securely and predictably delivered to protect your network with the best allowable quality within the bit rate budget.

VideoCharger can be integrated with Content Manager to enable search, archiving, management and sharing of rich digital assets, integrating them seamless into an enterprise content management infrastructure.

IBM Tivoli Storage Manager and IBM System Storage Archive Manager

In this chapter, we describe the IBM Tivoli Storage Manager software product and IBM System Storage Archive Manager. These products are the cornerstone on which IBM bases storage management. We explain how Tivoli Storage Manager provides an abstraction or virtualization layer between the storage of data and the management of the underlying storage devices. Also, we introduce the IBM System Storage Archive Manager (SSAM) and explain how it is different from a normal Tivoli Storage Manager system.

We discuss the following topics:

- ▶ IBM Tivoli Storage Manager and concepts
- ▶ Hierarchical Storage Management (HSM)
- ▶ IBM System Storage Archive Manager
- ▶ IBM Tivoli Storage Manager management of WORM storage devices
- ▶ Safeguarding IBM Tivoli Storage Manager
- ▶ SSAM and N series SnapLock

For additional information, refer to the IBM Redbooks, *Understanding the IBM System Storage DR550*, SG24-7091, and *IBM Tivoli Storage Management Concepts*, SG24-4877.

4.1 Tivoli Storage Manager concepts

Tivoli Storage Manager provides a comprehensive solution focused on the key data protection and management activities of backup, archive, recovery, space management, and disaster recovery (see Figure 4-1).

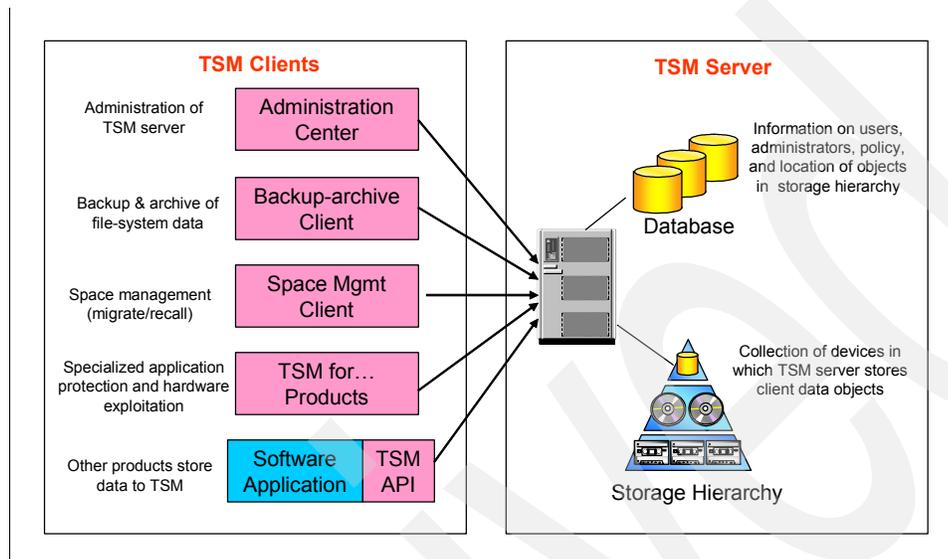


Figure 4-1 TSM architecture

Tivoli Storage Manager allows you to separate the backup, archiving, and retention of data from storage-related aspects of the data, in addition to many other services. Tivoli Storage Manager offers various storage management functions relevant to ILM:

- ▶ Data archiving defines how to insert data into the data retention system. Tivoli Storage Manager offers a command line interface to archive and back up files and a C language application programming interface (API) for use by content management applications.
- ▶ Data retention defines how long to keep the data object, not the individual tape. Tivoli Storage Manager offers various data retention options, such as these:
 - *By date* specifies the duration to retain the data.
 - *Event-based* determines retention on notification of a future event.
 - *Deletion hold* prevents deleting an object even after its defined retention period.
- ▶ Storage defines on which storage device to put the object. Tivoli Storage Manager supports hundreds of disk and tape storage devices and integrated hierarchical storage management of stored data. You can choose the most effective storage device for your requirements and subsequently let the data automatically migrate to different storage tiers.
- ▶ WORM functionality is offered by System Storage Archive Manager. The Tivoli Storage Manager administrator cannot accidentally or intentionally delete objects stored in Tivoli Storage Manager.
- ▶ Storage management services are provided by Tivoli Storage Manager. These additional storage management services facilitate hardware replacement and disaster recovery. Tivoli Storage Manager allows for easy migration to new storage devices when the old storage devices require replacing, and this is likely to happen when data is retained for long periods of time. Tivoli Storage Manager also offers functions to make multiple copies of archived data.

Tivoli Storage Manager offers a strong and comprehensive set of functions that you can exploit to effectively manage archived data. You can consider Tivoli Storage Manager an abstraction or virtualization layer between applications requiring data retention or storage management services and the underlying storage infrastructure.

4.1.1 Tivoli Storage Manager architectural overview

Tivoli Storage Manager is a client server software application that provides services such as network backup and archive of data to a central server. There are two main functional components in a Tivoli Storage Manager environment:

- ▶ You install the *Tivoli Storage Manager client* component on servers, computers, or machines that require Tivoli Storage Manager services. The Tivoli Storage Manager client accesses the data to be backed up or archived and is responsible for sending the data to the server.
- ▶ The *Tivoli Storage Manager server* is the central repository for storing and managing the data received from the Tivoli Storage Manager clients. The server receives the data from the client over the LAN network, inventories the data in its own database, and stores it on storage media according to predefined policies.

Figure 4-2 illustrates the components of a Tivoli Storage Manager environment. You can see that the core component is the Tivoli Storage Manager server.

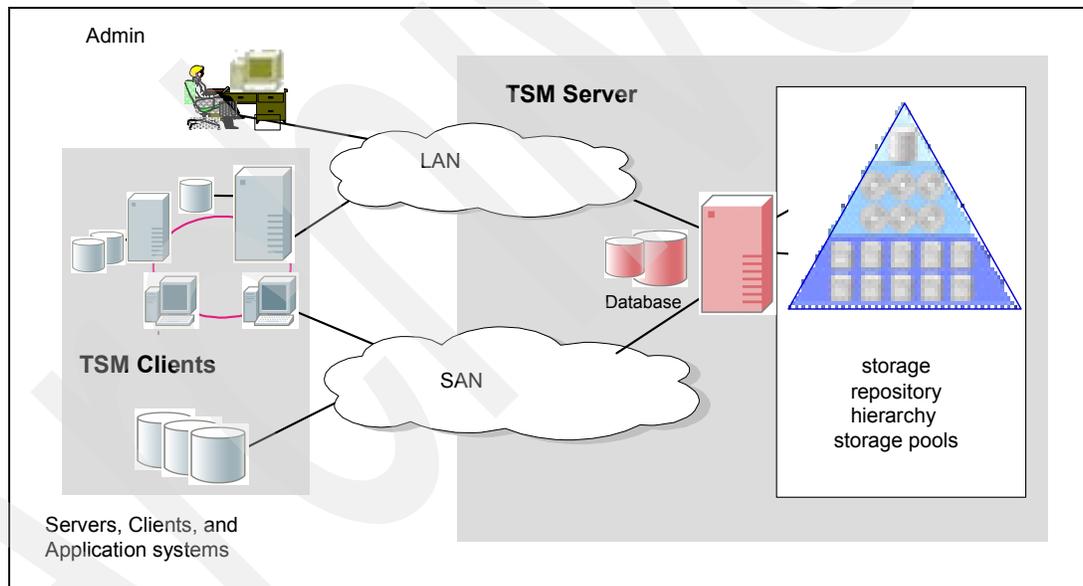


Figure 4-2 Tivoli Storage Manager components: architectural overview

We review and discuss the main components and functions of a Tivoli Storage Manager environment, emphasizing the components that are most relevant to an ILM-optimized environment. These components are:

- ▶ Tivoli Storage Manager server
- ▶ Administrative interfaces
- ▶ The server database
- ▶ Storage media management
- ▶ Data management policies
- ▶ Security concepts
- ▶ Backup Archive client interface
- ▶ Client application programming interface (API)

- ▶ Automation
- ▶ The client to server data path

Tip: For a detailed overview of Tivoli Storage Manager and its complementary products, refer to the following IBM Redbook:

<http://www.redbooks.ibm.com/abstracts/sg244877.html?open>

Tivoli Storage Manager server

The Tivoli Storage Manager server consists of a run-time environment and a relational database. You can install the server on several operating systems and on diverse hardware platforms, generally covering all popular environments. The proprietary database with its recovery log stores all the information about the current environment and the managed data. The Tivoli Storage Manager server listens for and communicates with the client systems over the LAN network.

Administrative interfaces

For the central administration of one or more Tivoli Storage Manager server instances, as well as the whole data management environment, Tivoli Storage Manager provides command line or Java-based graphical administrative interfaces, otherwise known as *administration clients* (see Figure 4-3).

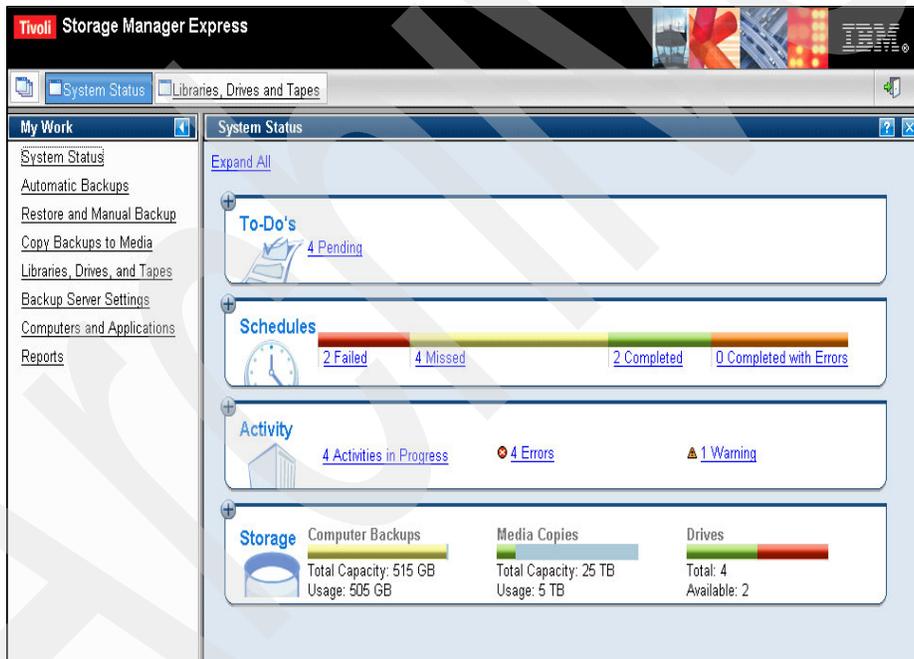


Figure 4-3 Administrative interface

The administrative interface enables administrators to control and monitor server activities, define management policies for clients, and set up schedules to provide services to clients at regular intervals.

The server database

The Tivoli Storage Manager server database is based on a relational database kernel that is integrated into and installed with the Tivoli Storage Manager server itself. The Tivoli Storage Manager server database stores all information relative to the Tivoli Storage Manager environment, such as the client nodes that access the server, storage devices, and policies. The Tivoli Storage Manager database contains one entry for each object stored in the Tivoli Storage Manager server, and the entry contains information, such as:

- ▶ Name of the object
- ▶ Tivoli Storage Manager client that sent the object
- ▶ Policy information or Tivoli Storage Manager *management class* associated with the object
- ▶ Location where the object is stored in the storage hierarchy

The Tivoli Storage Manager database retains information called *metadata*, which means data that describes data. The flexibility of the Tivoli Storage Manager database enables you to define storage management policies around business requirements for individual clients or groups of clients. You can assign client data attributes, such as the storage destination, number of versions, and retention period at the individual file level and store them in the database.

The Tivoli Storage Manager database also ensures reliable storage management processes. To maintain data integrity, the database uses a recovery log to roll back any changes made if a storage transaction is interrupted before it completes. This is known as a *two-phase commit*.

Also, you can mirror both the Tivoli Storage Manager database and recovery log for availability, providing automatic volume switching after a media failure. In the unlikely event of a Tivoli Storage Manager database recovery, operators can restore the database to the exact point of a failure by rolling the recovery log forward after restoring from the latest database backup.

Storage media management

Tivoli Storage Manager performs multiple diverse hierarchy and storage media management functions by moving or copying data between different pools or tiers of storage, as shown in Figure 4-4.

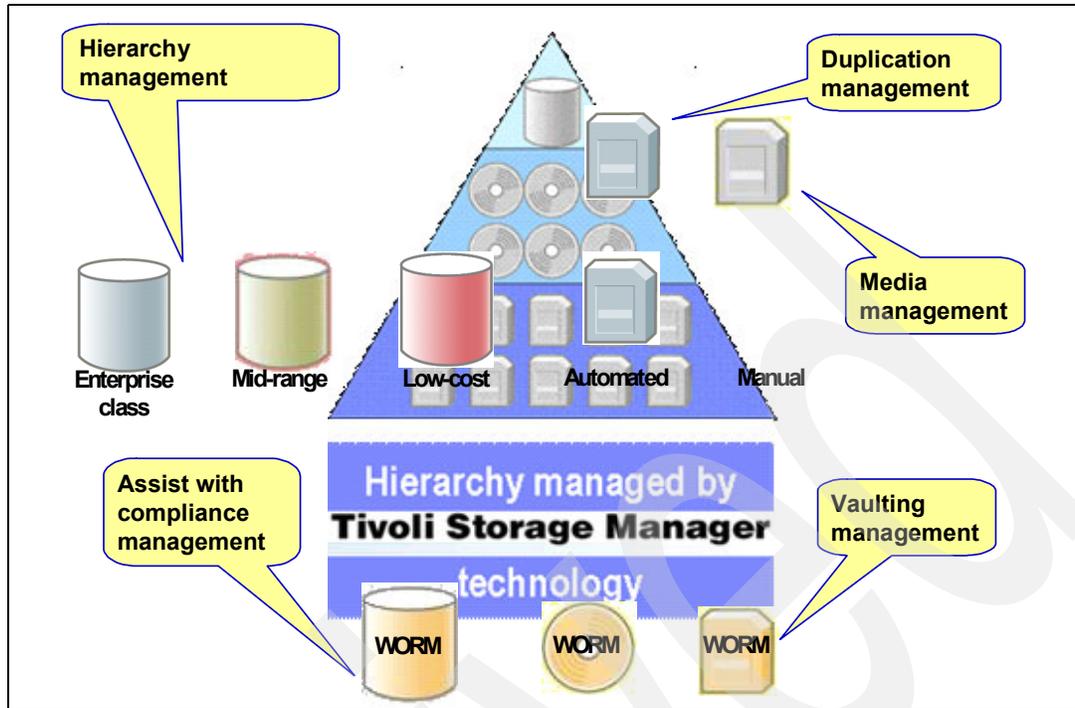


Figure 4-4 Tivoli Storage Manager management of the storage hierarchy

A Tivoli Storage Manager server can write data to more than 400 types of devices, including hard disk drives, disk arrays and subsystems, standalone tape drives, tape libraries, and other forms of random and sequential-access storage. The server uses media grouped into storage pools. You can connect the storage devices directly to the server through SCSI, through directly attached Fibre Channel, or over a Storage Area Network (SAN). Tivoli Storage Manager provides sophisticated media management capabilities that enable IT managers to perform the following tasks:

- ▶ Track multiple versions of files (including the most recent version)
- ▶ Respond to online file queries and recovery requests
- ▶ Move files automatically to the most cost-effective storage media
- ▶ Expire backup files that are no longer necessary
- ▶ Recycle partially filled volumes

Tivoli Storage Manager provides these capabilities for all backup volumes, including on-site volumes inside tape libraries, volumes that have been checked out of tape libraries, and on-site and off-site copies of the backups.

Tivoli Storage Manager provides a powerful media management facility to create multiple copies of all client data stored on the Tivoli Storage Manager server. Enterprises can use this facility to back up primary client data to two copy pools: One stored in an off-site location, and the other kept on-site for possible recovery from media failures. If a file in a primary pool is damaged or resides on a damaged volume, Tivoli Storage Manager automatically accesses the file from an on-site copy if it is available or indicates which volume should be returned from an off-site copy.

Tivoli Storage Manager also provides a unique capability for reclaiming expired space on off-site volumes without requiring the off-site volumes to be brought back on-site. Tivoli Storage Manager tracks the utilization of off-site volumes just as it does for on-site volumes. When the free space of off-site volumes reaches a determined reclamation threshold, Tivoli Storage Manager uses the on-site volumes to consolidate the valid files onto new volumes,

then directs the new volumes to be taken off-site. When the new tapes arrive off-site, Tivoli Storage Manager requests the return of the original off-site volumes, which can be reused as scratch volumes.

We discuss storage management in greater detail in Chapter 9, “Content Management and Integrated Storage Management” on page 217.

Data management policies

A data storage management environment consists of three basic types of resources: client systems, rules, and data. The client systems contain the data to manage, and the rules specify how the management must occur. For example, in the case of backup, how many versions you keep, where you store them, and so on (see Figure 4-5).

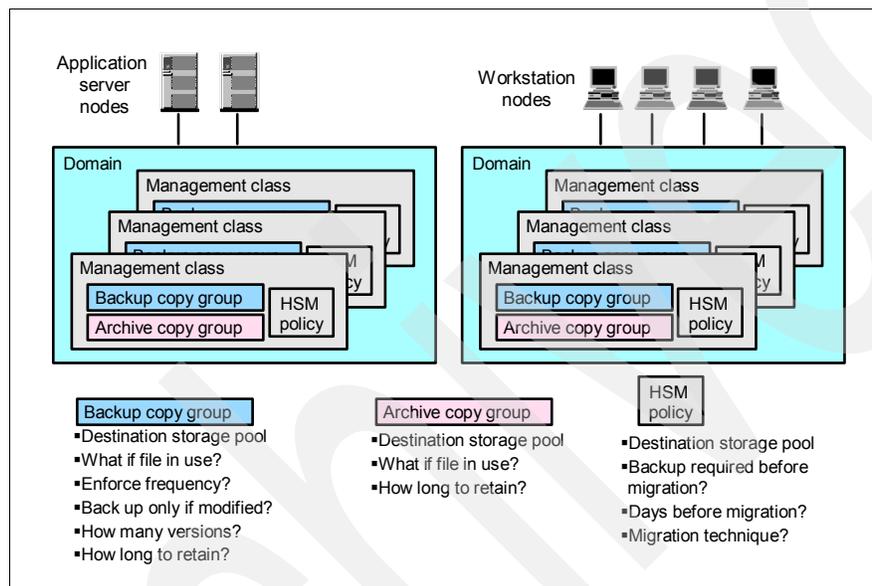


Figure 4-5 Policy Management

Tivoli Storage Manager policies define the relationships between these three resources. Depending on your actual requirements for managing your enterprise data, these policies can be simple or complex.

Tivoli Storage Manager has certain logical entities that group and organize the storage resources and define relationships between them. You group client systems, or nodes in Tivoli Storage Manager terminology, together with other nodes with common storage management requirements, into a policy domain.

We discuss these concepts in greater detail in 4.1.3, “Policy management” on page 85.

Security concepts

Because the storage repository of Tivoli Storage Manager is the place where an enterprise stores and manages all of its data, security is a vital aspect for Tivoli Storage Manager. To ensure that only the owning client or an authorized party can access the data, Tivoli Storage Manager implements, for authentication purposes, a mutual suspicion algorithm, which is similar to the methods used by Kerberos authentication.

Whenever a client (backup/archive or administrative) wants to communicate with the server, an authentication has to take place. This authentication contains both-sides verification, which means that the client has to authenticate itself to the server, and the server has to authenticate itself to the client.

To do this, all clients have a password, which is stored at the server side as well as at the client side. In the authentication dialog, these passwords are used to encrypt the communication. The passwords are not sent over the network, to prevent hackers from intercepting them. A communication session will be established only if both sides are able to decrypt the dialog. If the communication has ended, or if a time-out period has ended with no activity, the session will automatically terminate and a new authentication will be necessary.

Tivoli Storage Manager offers encryption of data sent by the client to the server. It offers both 128 bit AES and 56 bit DES encryption.

Backup Archive client interface

Tivoli Storage Manager is a client-server program. You must install the client product on the machine you want to back up. The client portion is responsible for sending and receiving data to and from the Tivoli Storage Manager server.

The Backup Archive client has two distinct features:

- ▶ The *backup feature* allows users to back up a number of versions of their data onto the Tivoli Storage Manager server and to restore from these, if the original files are lost or damaged. Examples of loss or damage are hardware failure, theft of computer system, or virus attack.
- ▶ The *archive feature* allows users to keep a copy of their data for long term storage and to retrieve the data if necessary. Examples of this are to meet legal requirements, to return to a previous working copy if the software development of a program is unsuccessful, or to archive files that are not currently necessary on a workstation.

The latter features are the central procedures around which Tivoli Storage Manager is built. Backup and archive are supporting functions to be able to retrieve lost data later on.

You can interact with the Tivoli Storage Manager server to run a backup/restore or archive/retrieve operation through three different interfaces:

- ▶ Graphical User Interface (GUI)
- ▶ Command Line Interface (CLI)
- ▶ Web Client Interface (Web Client)

The command line interface has a richer set of functions than the GUI. The CLI has the benefit of being a character mode interface, and, therefore, is well suited for users who have to type the commands. You might also consider using it when you cannot access the GUI interface or when you want to automate a backup or archive by using a batch processing file.

Client application programming interface (API)

Tivoli Storage Manager provides a data management application program interface (API) that you can use to implement application clients to integrate popular business applications, such as databases or groupware applications. The API also adheres to an open standard and is published to enable customers and vendors to implement specialized or custom clients for particular data management requirements or nonstandard computing environments.

The Tivoli Storage Manager API enables an application client to use the Tivoli Storage Manager storage management functions. The API includes function calls that you can use in an application to perform the following operations:

- ▶ Start or end a session
- ▶ Assign management classes to objects before they are stored on a server
- ▶ Archive objects to a server
- ▶ Signal retention events for retention, such as activate, hold, or release

Alternatively, some vendor applications exploit the Tivoli Storage Manager data management API by integrating it into their software product itself to implement new data management functions or to provide archival functionality on additional system platforms. Some examples are IBM DB2 Content Manager, IBM DB2 Content Manager OnDemand, IBM CommonStore for SAP R/3, Lotus Domino, and Microsoft Exchange data archival.

The API, including full documentation available on the Internet, is published to enable customers and vendors to implement their own solutions to meet their requirements. For more information, see *IBM Tivoli Storage Manager: Using the Application Program Interface*, GC32-0793, available at:

http://publib.boulder.ibm.com/infocenter/tivihelp/v1r1/topic/com.ibm.itsmc.doc/ans_a0000.pdf

Automation

Tivoli Storage Manager includes a central scheduler that runs on the Tivoli Storage Manager server and provides services for use by the server and clients. You can schedule administrative commands to tune server operations and start functions that require significant server or system resources during times of low usage. You can also schedule client action, although unusual for a data retention-enabled client. Each scheduled command action (administrative or client) is called an *event*. The server tracks and records each scheduled event and its completion status in the Tivoli Storage Manager server database.

Client to server data path

Tivoli Storage Manager data can travel from client to server either over the LAN network or the SAN network when using Tivoli Storage Manager for SAN to enable LAN-free data transfers. The diagram in Figure 4-6 schematically illustrates the components and data paths in a Tivoli Storage Manager environment.

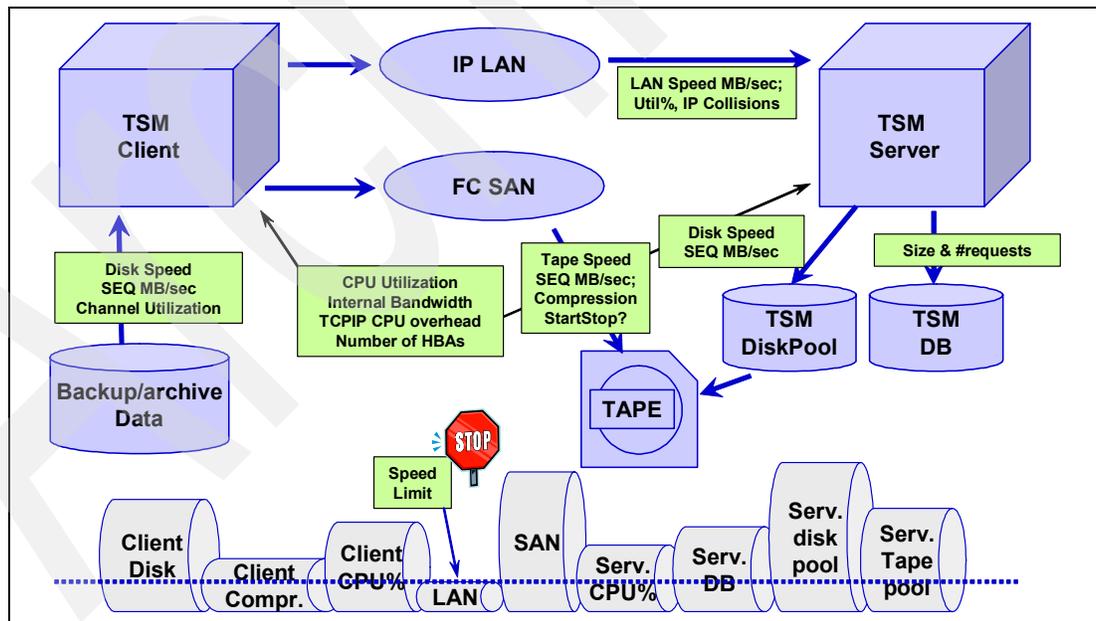


Figure 4-6 Backup environment pipeline and data flows

Figure 4-6 shows the data flow or pipeline and potential bottlenecks in a Tivoli Storage Manager environment. It illustrates the route the data takes through the many components of the client-server storage environment. For each step in this route, we list causes of potential performance bottlenecks.

Data is read by the backup or archive client from client disk or transferred in memory to the API client from a content manager application. The Tivoli Storage Manager client, depending on the options set, can compress the data before sending it to the Tivoli Storage Manager server in order to reduce network utilization.

The client has the option to use the LAN or the SAN, also called LAN-free, for data transport. The SAN is optimized for bulk transfers of data and allows writing directly to the storage media, bypassing the Tivoli Storage Manager server and the network. LAN-free support requires an additional IBM Tivoli Storage Manager license called IBM Tivoli Storage Manager for SAN. Archiving data is normally a low volume operation, handling relatively small amounts of data to be retained (see Figure 4-7).

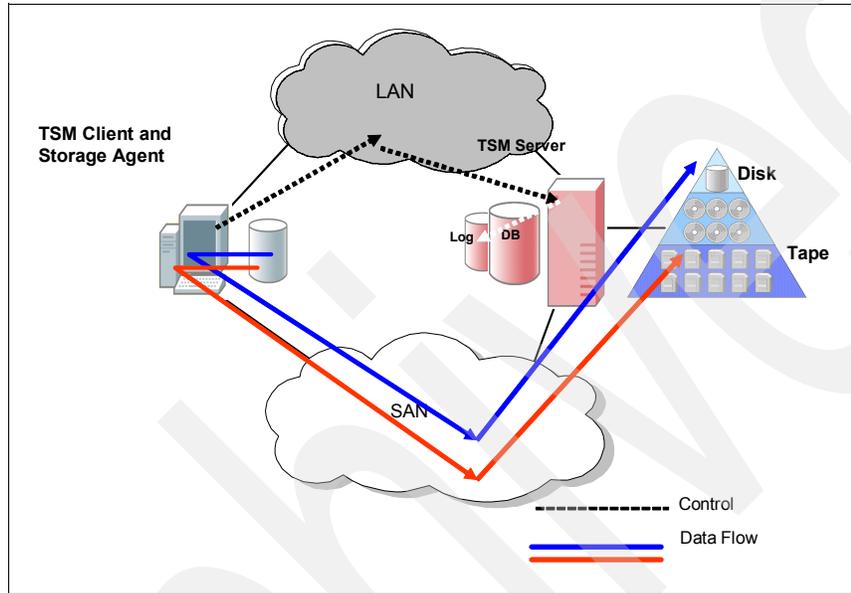


Figure 4-7 LAN free

The Tivoli Storage Manager server receives metadata, and data when using LAN transport, over the LAN network. Tivoli Storage Manager then updates its database. Many small files potentially can cause a high level of database activity.

When the data is received over the LAN, it generally is stored in a disk storage pool for later migration to tape as an overflow location.

The maximum performance of data storage or retrieval operations depends on the slowest “link in the chain”, another way of illustrating it is that performance is constrained by the smallest pipe in the pipeline, as shown in Figure 4-6. In the figure, the LAN is the constraint on performance.

4.1.2 Tivoli Storage Manager storage management

Tivoli Storage Manager manages client *data objects* based on information provided in administrator-defined *policies*.

Data objects can be subfile components, files, directories, or raw logical volumes that are archived from client systems. They can be objects such as tables, logs, or records from database applications, or simply a block of data that an application system archives to the server. The Tivoli Storage Manager server stores these objects on disk volumes and tape media that it groups into *storage pools*.

Tivoli Storage Manager storage pools and storage hierarchy

Tivoli Storage Manager manages data as objects as they exist in Tivoli Storage Manager storage pools. See Figure 4-8.

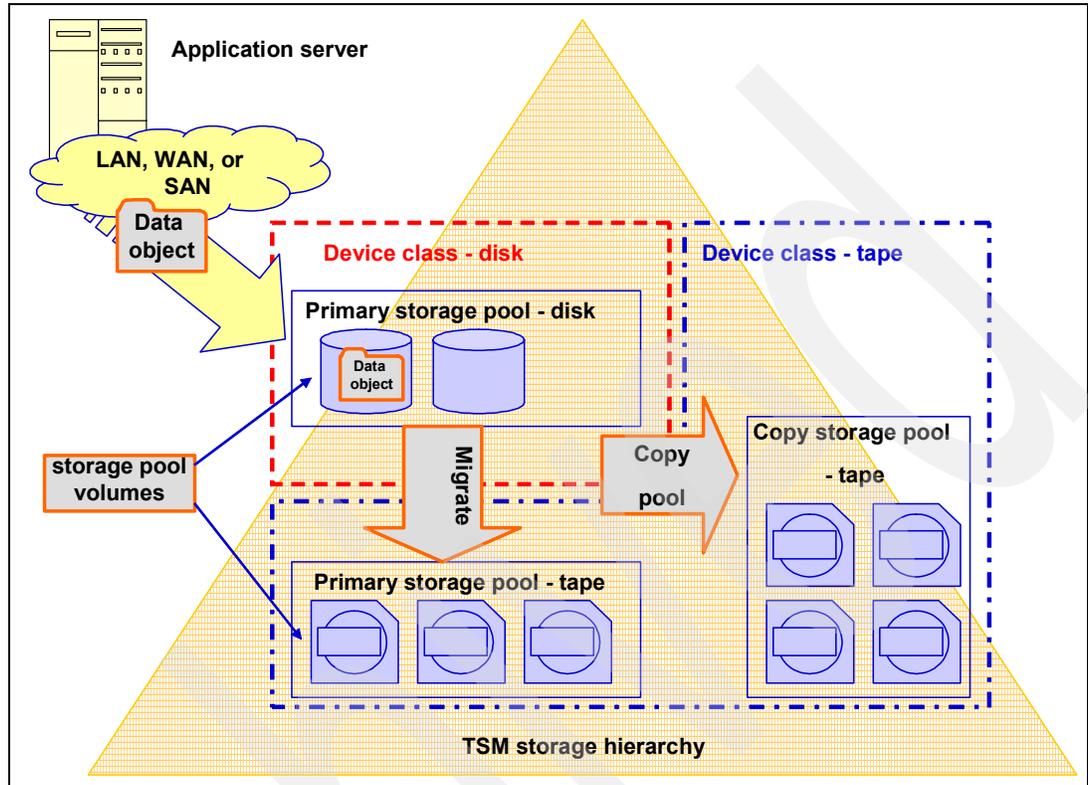


Figure 4-8 Tivoli Storage Manager storage hierarchy

Each object is “bound” to an associated management policy. The policy defines how long to keep that object and where the object enters the storage hierarchy.

The physical location of an object within the storage pool hierarchy has no effect on its retention policies. You can migrate or move an object to another storage pool within a Tivoli Storage Manager storage hierarchy. This can be useful when freeing up storage space on higher performance devices, such as disk, or when migrating to new technology.

You can and should also copy objects to copy storage pools. To store these data objects on storage devices and to implement storage management functions, Tivoli Storage Manager uses logical definitions to classify the available physical storage resources. Most important is the logical entity called a *storage pool*, which describes a storage resource for a single type of media, such as disk volumes, which are files on a file system, or tape volumes, which are cartridges in a library.

Device classes

A storage pool is built up from one or more Tivoli Storage Manager storage pool volumes. For example, a disk storage pool can consist of several AIX raw logical volumes or multiple AIX files on a file system. Each AIX raw logical volume or AIX file corresponds to one Tivoli Storage Manager storage pool volume.

A logical entity called a *device class* is used to describe how Tivoli Storage Manager can access those physical volumes to place the data objects on them. Each storage pool is bound to a single device class.

The storage devices used with Tivoli Storage Manager can vary in their technology and total cost. To reflect this fact, you can imagine the storage as a pyramid (or triangle), with high-performance storage in the top (typically disk), normal performance storage in the middle (typically optical disk or cheaper disk), and low-performance, but high-capacity, storage at the bottom (typically tape). Figure 4-8 illustrates this tiered storage environment that Tivoli Storage Manager uses:

- ▶ Disk storage devices are random access media, making them better candidates for storing frequently accessed data. Disk storage media with Tivoli Storage Manager can accept multiple parallel data write streams.
- ▶ Tape, however, is an economical high-capacity sequential access media, which you can easily transport off-site for disaster recovery purposes. Tape is recommended for large files so that the data streaming capabilities of tape drive technology can be exploited.

Disk storage is referred to as online storage, while tape storage has often been referred to as *off-line* and also *near-line* with regard to Hierarchical Storage Management (HSM) in the past. With Tivoli Storage Manager for Space Management, tape volumes, located in a tape library, are accessed by the application that is retrieving data from them (near-line) transparently. Tapes no longer in the library are off-line, requiring manual intervention. The introduction of lower cost mass storage devices, such as Serial Advanced Technology Attachment (SATA) disk systems, offers an alternative to tape for near-line storage. Figure 4-9 illustrates the use of a SATA disk as near-line storage.

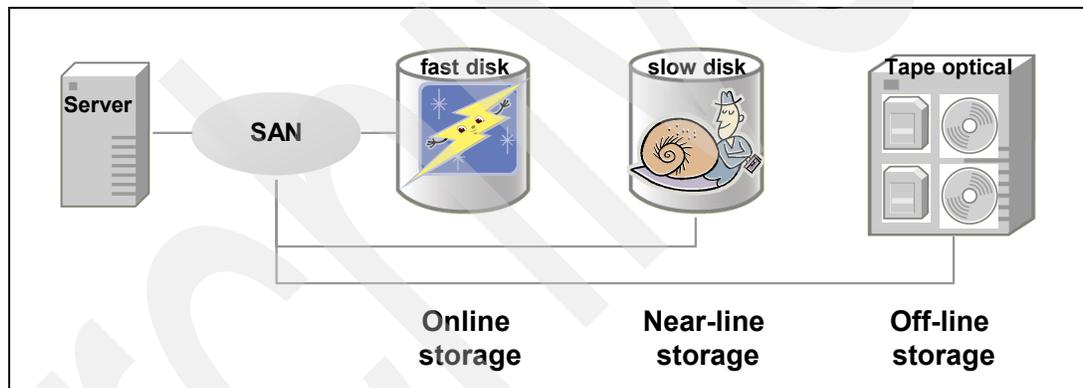


Figure 4-9 Online, near-line, and off-line storage

Device types

Each device defined to Tivoli Storage Manager is associated with one device class. Each device class specifies a *device type*. A device type identifies a device as a member of a group of devices, devices that shares similar media characteristics. For example, the 3592 device type applies to IBM TotalStorage Enterprise Tape Drive 3592.

The device type also specifies management information, such as how the server gains access to the physical volumes, recording format, estimated capacity, and labeling prefixes.

Device types include DISK, FILE, and a variety of removable media types for tape and optical devices. Note that a device class for a tape or optical drive must also specify a *library*. The *library* defines how Tivoli Storage Manager can mount a storage volume onto a storage device such as a tape drive.

Device access strategy

The access strategy of a device is either *random* or *sequential*. Primary storage pools can use random devices (such as disk) or sequential devices (such as tape). Copy storage pools use sequential access devices. Certain Tivoli Storage Manager processes use only sequential access strategy device types:

- ▶ Copy storage pools
- ▶ Tivoli Storage Manager database backups
- ▶ Export
- ▶ Import

Tape devices

Tivoli Storage Manager supports a wide variety of enterprise class tape drives and libraries. The following link connects you to the product support Web site where you can find information about the currently supported devices:

http://www.ibm.com/software/sysmgmt/products/support/IBM_TSM_Supported_Devices_for_AIXHPSUNWIN.html

We recommend that you use tape devices for backing up your primary storage pools to copy storage pools and for backing up the database. Tape devices are well suited for this, because the media can be transported off-site for disaster recovery purposes.

4.1.3 Policy management

A data storage management environment consists of three basic types of resources: *client system*, *policy*, and *data*.

The client systems run the applications that create or collect data to manage, for example, applications using the API to archive data.

The policies are the rules to specify how to manage the archive objects. For example, how long to retain an archive object in storage; whether chronological or event-based archive retention is used; in which storage pool to place an object, or, in the case of backup, how many versions to keep, where they should be stored, and what Tivoli Storage Manager does to the archive object after the data is no longer on the client file system.

Client systems, or *nodes*, in Tivoli Storage Manager terminology, are grouped together with other nodes with common storage management requirements into a *policy domain*. The policy domain links the nodes to a *policy set*, a collection of storage management rules for different storage management activities.

Note: The term *client node* refers to the application sending data to the Tivoli Storage Manager server.

A policy set consists of one or more *management classes*. A management class contains the rule descriptions called *copy groups* and links these to the data objects to manage. A copy group is the place where you define all the storage management parameters, such as the number of stored copies, retention period, and storage media. When the data is linked to particular rules, it is said to be bound to the management class that contains those rules.

Another way to look at the components that make up a policy is to consider them in the hierarchical fashion in which they are defined; that is, consider the policy domain containing the policy set, the policy set containing the management classes, and the management classes containing the copy groups and the storage management parameters, as illustrated in Figure 4-10.

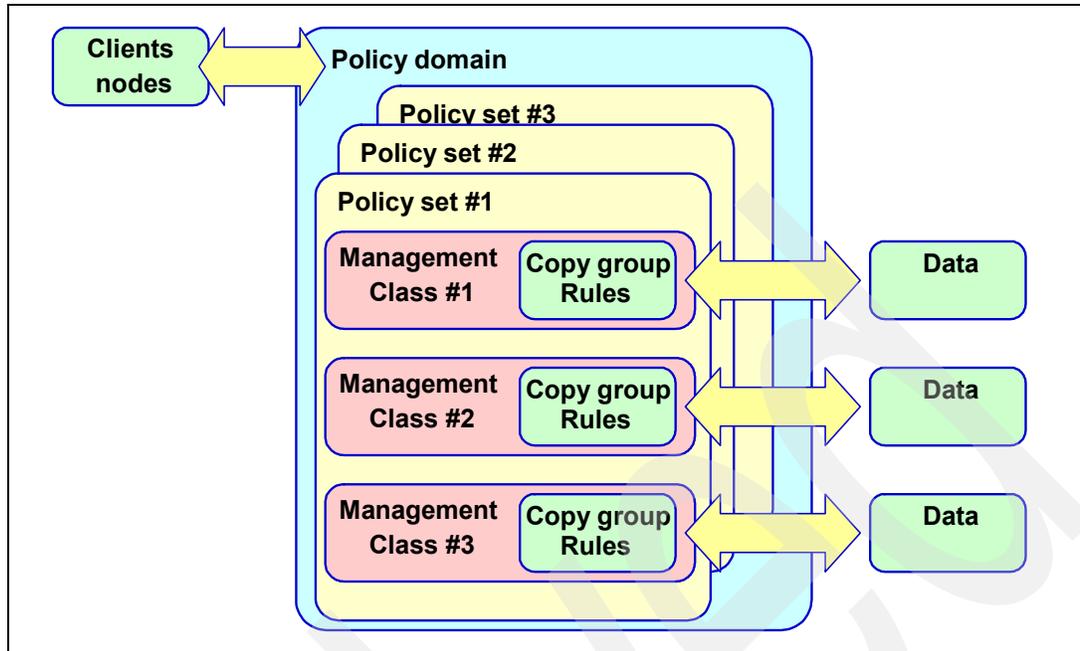


Figure 4-10 Policy relationships and resources

We explain the relationship between the items in Figure 4-10 in the following pages.

Copy group rules

Copy group rules can define either a backup copy group or an archive copy group. One set of rules applies to backups and a separate set to archives.

Backup copy group

This copy group controls the backup processing of files associated with the specific management class. It is uncommon to use backup copy groups for archival or data retention applications because they are better suited to backup versioning of files. A backup copy group determines:

- ▶ Where to store the object
- ▶ What to do if the file is in use
- ▶ Whether or not to backup only if modified or changed
- ▶ Enforce minimum frequency of backup, to avoid backing up every time
- ▶ If the file exists on the client node:
 - How many copies to keep
 - How long to keep them
- ▶ If the file has been deleted on the client:
 - How many copies to keep
 - How long to keep the last copy of the file

Archive copy group

This copy group controls the archive processing of files associated with the management class. An archive copy group determines:

- ▶ How the server handles files that are in use during archive
- ▶ Where the server stores archived copies of files
- ▶ How long the server keeps archived copies of files

Management class

The management class associates client files with archive copy groups with files. A management class is a Tivoli Storage Manager *policy*.

Each individual object stored in Tivoli Storage Manager is associated with one and only one management class. A management class is a container for copy groups; it can contain either one backup or archive copy group, both a backup and an archive copy group, or no copy groups at all. Users can bind (that is, associate) their files to a management class through the *include-exclude list*, a set of statements or rules that associate files to a management class based on file filtering rules. Alternatively, a user can explicitly request an archive management class.

Policy set

The policy set specifies the management classes that are available to groups of users. Policy sets contain one or more management classes. You must identify one management class as the default management class. Only one policy set, the ACTIVE policy set, controls policies in a policy domain.

Policy domain

The concept of policy domains enables an administrator to group client nodes by the policies that govern their files and by the administrators who manage their policies. A policy domain contains one or more policy sets, but only one policy set (named ACTIVE) can be active at a time. The server uses only the ACTIVE policy set to manage files for client nodes assigned to a policy domain.

You can use policy domains to:

- ▶ Group client nodes with similar file management requirements.
- ▶ Provide different default policies for different groups of clients.
- ▶ Direct files from different groups of clients to different storage hierarchies based on requirements.
- ▶ Restrict the number of management classes to which clients have access.

Figure 4-11 summarizes the relationships among the physical device environment, Tivoli Storage Manager storage and policy objects, and clients. The numbers in the following list correspond to the numbers in the figure.

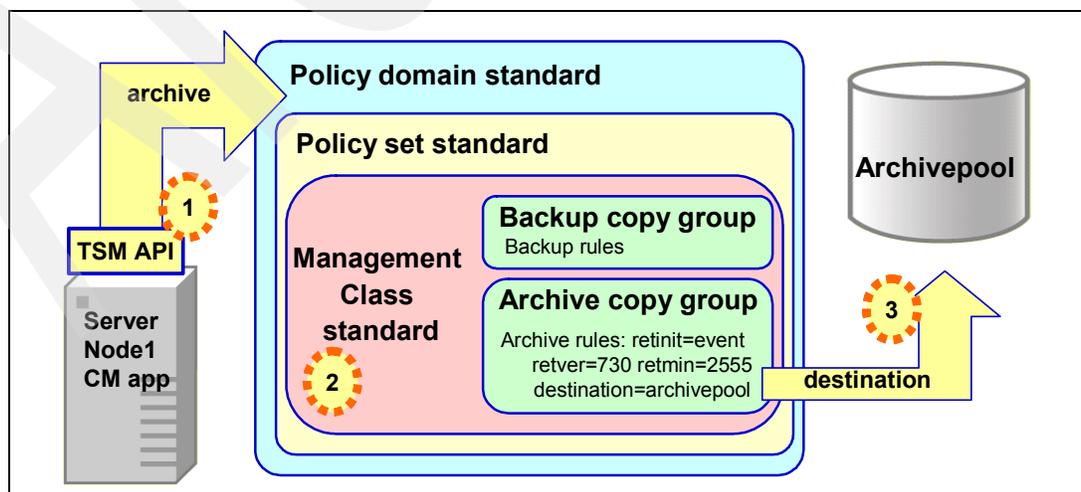


Figure 4-11 Basic policy structure for archive

Figure 4-11 shows an outline of the policy structure. These are the steps to create a valid policy:

1. When clients are registered, they are associated with a policy domain. Within the policy domain are the policy set, management class, and copy groups.
2. When a client (application) archives an object, the object is bound to a management class. A management class and the archive copy group within it specify where files are stored first (destination), and how they are managed when they are archived.
3. Storage pools are the destinations for all stored data. An archive copy group specifies a destination storage pool for archived files. Storage pools are mapped to device classes, which represent devices. The storage pool contains volumes of the type indicated by the associated device class.

Data stored in disk storage pools can be migrated to tape or optical disk storage pools and can be backed up to copy storage pools.

4.2 Hierarchical storage management

Hierarchical storage management (HSM) refers to a function of Tivoli Storage Manager that automatically distributes and manages data on disk, tape, or both by regarding devices of these types and potentially others as levels in a storage hierarchy. The devices in this storage hierarchy range from fast, expensive devices to slower, cheaper, and possibly removable devices. The objectives are to minimize access time to data and maximize available media capacity.

Hierarchical storage management is implemented in many IBM products, such as Tivoli Storage Manager, in System i™, and in z/OS in the combination of the storage management subsystem (SMS), DFSMSHsm™, DFSMSdss™, and DFSMSrmm™.

Tivoli Storage Manager for Space Management solutions are applied to data on storage media, such as disk. The data is automatically migrated from one level of storage media to the next level based on some predefined policy. Tivoli Storage Manager offers different kinds of HSM functionality.

4.2.1 HSM in the Tivoli Storage Manager server

One level of HSM is related to how the Tivoli Storage Manager server stores data — that is, on storage pools or collections of storage volumes of the same media type, as discussed in 4.1.2, “Tivoli Storage Manager storage management” on page 82. You can map different Tivoli Storage Manager storage pools to different device types, and they can be concatenated together into a hierarchy using the Tivoli Storage Manager `nextstgpool` parameter.

Figure 4-12 illustrates a Tivoli Storage Manager server hierarchy with three storage pools. Storage pools are managed by threshold. Each pool has a high threshold and a low threshold. When the amount of data in the storage pool exceeds the high threshold, Tivoli Storage Manager initiates a migration process to move the data.

The data is moved to a destination called *next storage* pool, which is defined as a storage pool parameter in the original storage pool. Therefore, in the example we see that poolfast has a next storage pool called poolslow. The migration process will move data from poolfast to poolslow. The process starts when the amount of data stored in poolfast exceeds the high migration threshold and stops when it reaches the low threshold.

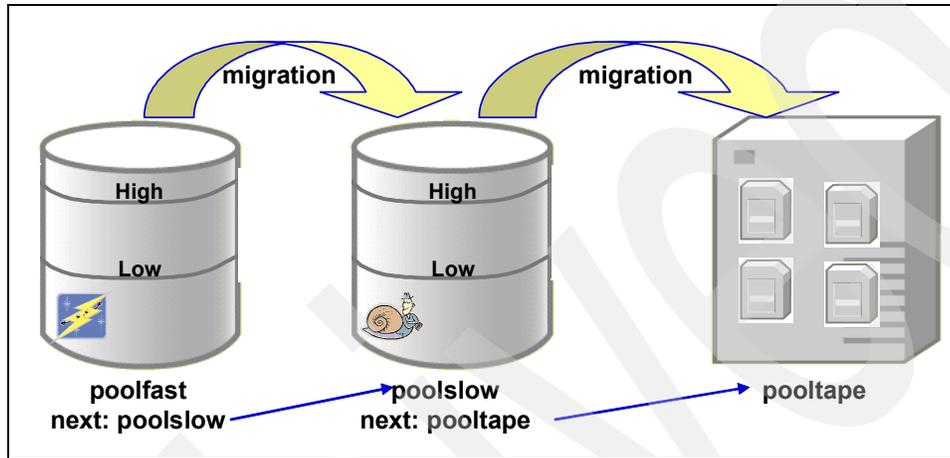


Figure 4-12 Tivoli Storage Manager server migration processing

Tivoli Storage Manager offers additional parameters to control migration of data from one storage pool to the next. One of these is *migdelay*, which specifies the minimum number of days that a file must remain in a storage pool before the file becomes eligible for migration to the next storage pool.

4.2.2 Space management for file systems

Tivoli Storage Manager offers two separate space management clients for file systems: one for UNIX® and one for Windows environments.

In both cases, the space management client resides on the file server where you want to perform space management. It moves files from the local file system to lower cost storage managed by the Tivoli Storage Manager server, and this movement is called *migration*. Tivoli Storage Manager performs this movement based on criteria such as file size and age.

Moving a file to the Tivoli Storage Manager server implies that the file is removed from the Tivoli Storage Manager client. The client file system continues to see the file as though it were still on local disk. When a request to access the file occurs, the space management client intercepts the file system requests and, depending on operating system platform, either *recalls* the file to primary storage or, in some cases, can redirect the file system request to secondary storage. These operations are performed transparently to the file system request even though the request can be slightly delayed because of the tape mount processing.

Figure 4-13 Illustrates a sample HSM storage hierarchy built to minimize storage costs.

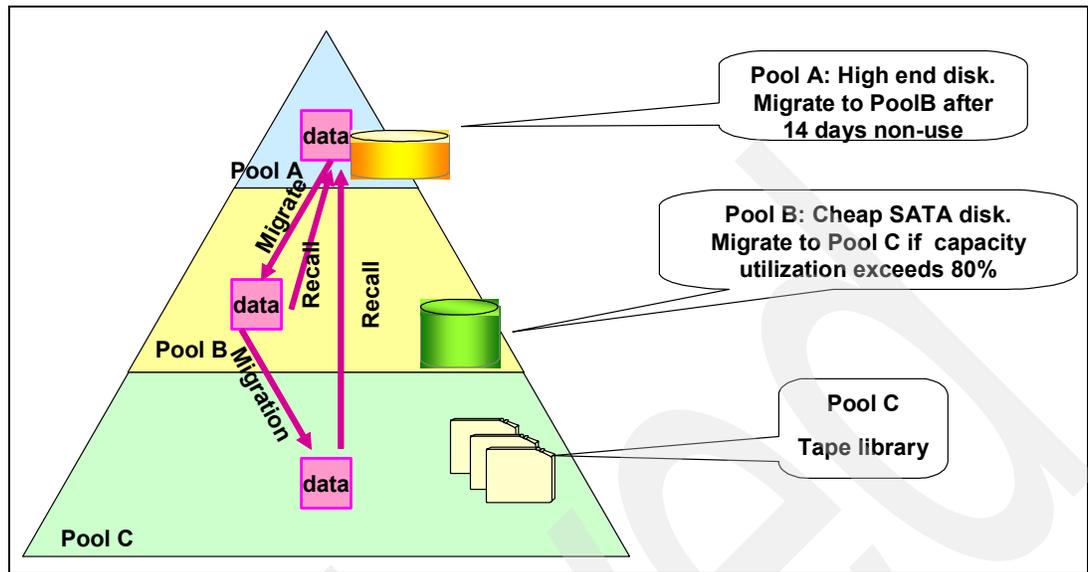


Figure 4-13 Sample cost-based HSM storage hierarchy

Space management for UNIX clients

The Tivoli Storage Manager for Space Management for UNIX client migrates files from your local file system to storage and recalls them either automatically or selectively. Migrating files to a distributed storage device frees space for new data on your local file system.

Your Tivoli Storage Manager administrator defines management classes to files. You, as root user, can:

- ▶ Select space management options and settings.
- ▶ Assign management classes to your files.
- ▶ Exclude files from space management.
- ▶ Schedule space management services.

These options and settings determine which files are eligible for automatic migration, the order in which files are migrated, where the migrated files are stored, and how much free space is maintained on your local file system. You prioritize files for migration by their file size, or by the number of days since your files were last accessed. Stub files that contain the necessary information to recall your migrated files remain on your local file system so that the files appear to reside locally. When you access migrated files, they are recalled automatically to your local file system. This is different from archiving, which completely removes files from your local file system.

The Space Management client provides space management services for locally mounted file systems, and it migrates regular files only. It does not migrate character special files, block special files, named pipe files, or directories.

File migration, unlike file backup, does not protect against accidental file deletion, file corruption, or disk failure. Continue to back up your files whether they reside on your local file system or in Tivoli Storage Manager storage. You can use the Tivoli Storage Manager backup-archive client to back up and restore migrated files in the same manner as you would back up and restore files that reside on your local file system. If you accidentally delete stub files from your local file system, or if you lose your local file system, you can restore the stub files from Tivoli Storage Manager.

For planned processes, such as storing a large group of files in storage and returning them to your local file system for processing, use the archive and retrieve processes. You can use the backup-archive client to archive and retrieve copies of migrated files in the same manner as you would archive and retrieve copies of files that reside on your local file system.

Space Management supports various file systems. Currently, these integrations exist:

- ▶ File system proprietary integration:
Data can be directly accessed and read from any tier in the storage hierarchy. This is supported on JFS on AIX.
- ▶ DMAPI standard-based integration:
The Data Management Application Programming Interface (DMAPI) standard has been adopted by several storage management software vendors. File system vendors focus on the application data management part of the protocol. Storage management vendors focus on the hierarchical storage management part of the protocol. Tivoli Storage Manager for Space Management Client supported platforms currently are: GPFS on AIX, VxFS on Solaris, GPFS on xLinux, and VxFS on HP.

Space management for Windows clients

IBM offers HSM functionality on windows with the Tivoli Storage Manager for Space Management for Windows client, starting with Version 5.3.

Space Management for Windows offers automated management features, such as:

- ▶ Policy-based file selection to apply Space Management rules to predefined sets of files
- ▶ On-demand scheduling to define when to perform Space Management automatic archiving
- ▶ Transparent recall to automatically have an application to reference a migrated file

The policies or rules that Space Management for Windows supports allow you to filter files based on attributes, such as:

- ▶ Directory name
- ▶ File types, based on the extensions
- ▶ Creation, modification, or last access date of file

Automatic archiving performs archiving operations based on inclusion or exclusion of directories and subdirectories and inclusion or exclusion of file extensions. In addition, you can configure filter criteria based on creation, modification, and last access date.

Note: Space Management for Windows uses the term *automatic archiving* to indicate the migration of a file to the Tivoli Storage Manager server.

You can configure automatic archiving to occur on a periodic basis. This could be daily, weekly, or monthly. Automatic archiving can be controlled via the Windows task scheduler or any other scheduler tool such as IBM Tivoli Workload scheduler.

What happens to the original file on the Windows file server after archiving depends on the *archiving mode*, which you can define for each archiving job. You can keep the original file, replace the file with a *shortcut*, or delete the file. Replacing the original file with a shortcut is the most common option. Replacing a file with a shortcut means that the original file is replaced by a sparse file that preserves the original file attributes. In addition, a Windows Reparse Point is generated identifying an archived file and containing the data required to reload the file from the archive.

Windows Explorer will continue to report the logical (original) size of the file. The icon of a shortcut is the original. A small clock in the icon indicates that the actual file is stored on remote storage.

To open an archived document in Windows Explorer, you simply click the file icon. The retrieval occurs in the background without any further action. The retrieval happens in a fully transparent and synchronous mode.

A retrieved document is not automatically removed from the back-end repository. Retrieval does not affect the archived object. This is important for compliance. You can regard a retrieved file as a temporary copy in the file system.

When you modify a retrieved document, Space Management for Windows recognizes the modification and stores the modified file as a new version in the archive during execution of the next archiving job. For read and restore operations, the most recent version of the file is always accessed.

4.3 System Storage Archive Manager

Policy-based data retention and disposition has long been recognized as a storage administrator's tool for efficiently managing storage resource utilization. However, in today's regulatory and potentially litigious environment, policy-based data retention and disposition is recognized as a must for records and information management.

Here, we explain what IBM System Storage Archive Manager (SSAM) is, and how it integrates with storage hardware.

IBM System Storage Archive Manager (SSAM) is a version of Tivoli Storage Manager that has Data Retention Protection enabled. This function ensures that objects that have been archived will not be deleted from the Tivoli Storage Manager server until the retention policies set for that object have been satisfied. SSAM actively inhibits deletion of unexpired objects.

4.3.1 Reasons for data retention

In the last few years there has been a growing business focus on data retention for compliance reasons and data disposition at the end of the retention period. We can outline some trends:

- ▶ Data, both file and individual record, retention policies are defined by regulations, laws, and corporate policies. Data must be retained for the period defined by the regulation.
- ▶ Data must be *discoverable*, so that when it is required it can be searched and easily found again. There is a requirement for inventory and indexing tools to manage all this data so as to make it discoverable.
- ▶ Data that is *discoverable*, beyond the required retention policy, can be used as legal evidence. Discoverable data refers to data that has passed its retention date and has not yet been deleted. Data destruction at or shortly after the point of disposition is desirable in order to avoid possible legal evidence.
- ▶ Low cost, high capacity Serial Advanced Technology Attachment (SATA) disk storage is displacing optical storage for long-term retained data. SATA disk technology is an additional option to tape for backup data storage.

- ▶ Automated storage management functions are required to effectively manage the growth of reference data. Reference data is data that has to be managed for retention and compliance reasons. Examples of such services are:
 - Storage resource management
 - Tiered storage management
 - Hierarchical storage management
 - Storage technology migration
 - Backup/recovery and disaster recovery

Data access requirements can vary over its lifetime; generally, archival data decreases in importance as time goes by. Sporadically, the data becomes important again when it is accessed for regulatory or application requests. Figure 4-14 shows the lifecycle requirements, access speed, and frequency over time. As time passes, frequency of access normally decreases, and data can be automatically moved to more cost-effective storage.

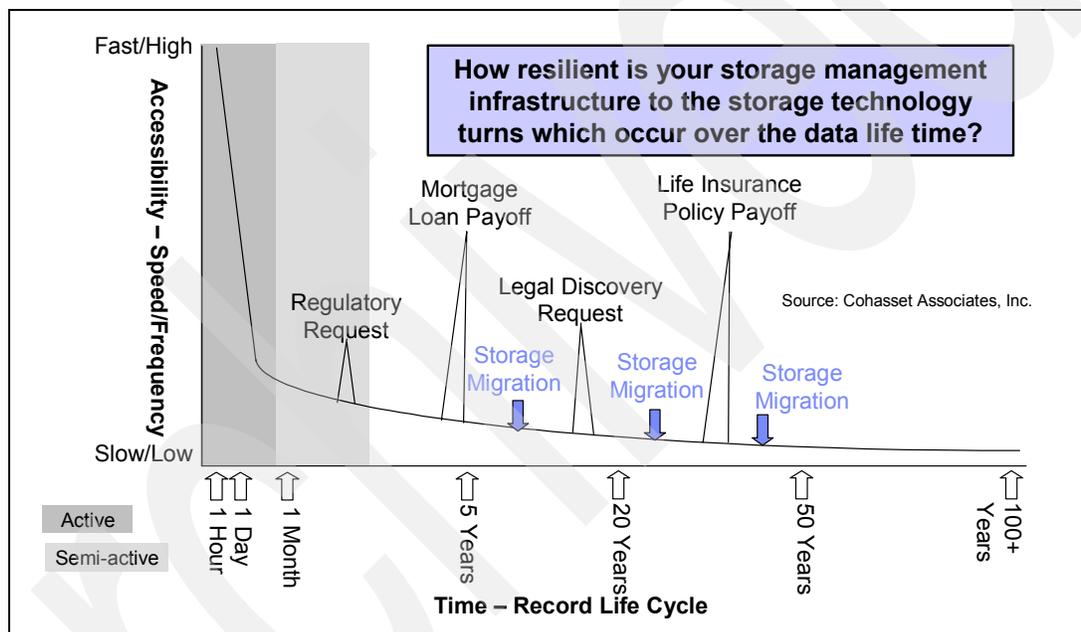


Figure 4-14 Data access requirements over retention period

Data retention functions and solutions offer important advantages. The up-front costs of data retention solutions might seem steep, but consider the potential costs and liabilities of not having a robust solution. In the following pages, we discuss common reasons for developing such a solution.

Data retention solutions help find useful documentation necessary during litigation. The cost of managing, searching, and retrieving old data can be very high. Often companies try to settle litigation out of court simply to avoid costly retrievals of data or because it is virtually impossible for them to retrieve the data.

A second benefit of data retention solutions is to avoid retaining unnecessary data that could be used against a company: Potentially damaging data can cost a company millions of dollars.

Data retention provides a solution that allows for the movement of the data to new technology when it comes available. This ensures lifecycle management of data and removes the necessity and cost of keeping old and obsolete technology around.

Data retention solutions based on Tivoli Storage Manager allow for the transparent migration of data between different tiers inside the Tivoli Storage Manager server storage repository, using standard Tivoli Storage Manager migration functions. As the data ages, it automatically moves down the tiers of the storage hierarchy in the Tivoli Storage Manager server.

Data retention allows for the efficient reuse of storage, improving your Return on Investment (ROI). After data can be deleted, you can expire it and reuse the space. Alternatively the data can be automatically and transparently migrated to more cost-effective storage devices.

Some companies have stated that they have *certified* applications for particular regulations. However, this is just marketing hype. There are no certification processes for data retention solutions of which Cohasset Associates is aware¹. Most customers' legal departments are simply cautious and use procedures and hardware that others in the industry are following. Figure 4-15 illustrates the electronic chain of trust between the application and the underlying storage device.

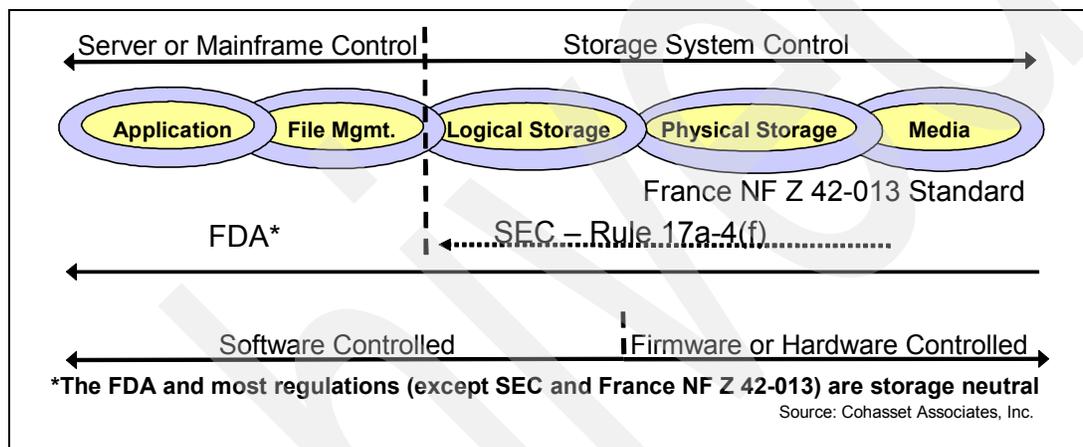


Figure 4-15 Electronic records chain of trust

In order for Tivoli Storage Manager to meet the requirements of some of the regulations, we had to add the ability to do retention protection to our already robust archive solution, as illustrated in Figure 4-16. This version of Tivoli Storage Manager is called the System Storage Archive Manager (SSAM).

¹ Cohasset Associates, White Paper for IBM, October 2004

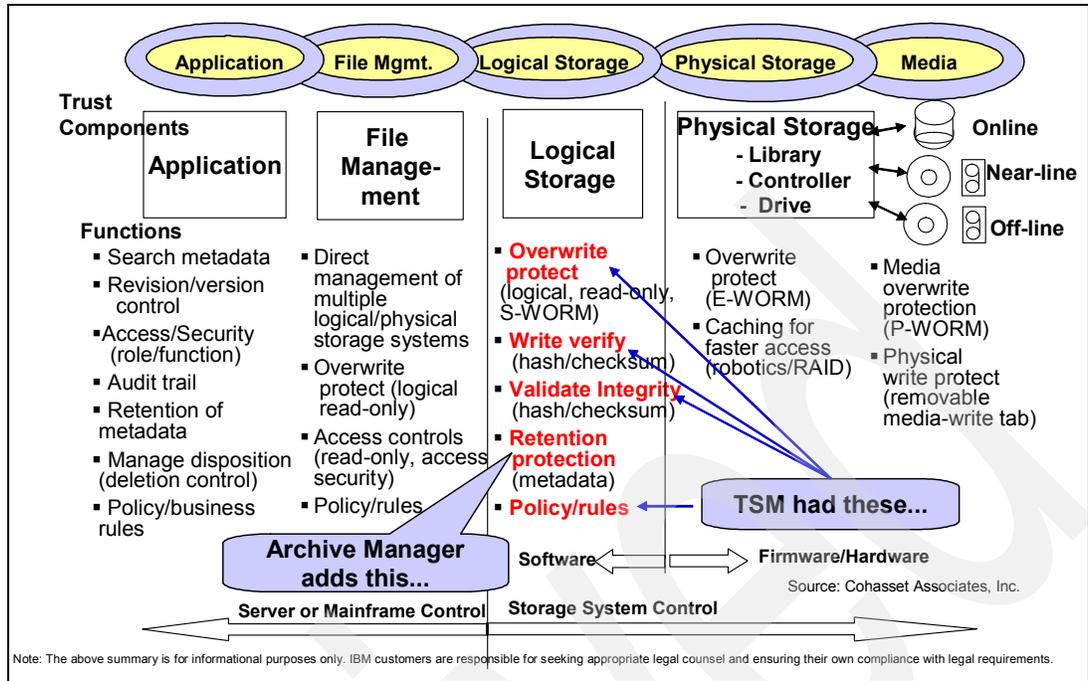


Figure 4-16 SSAM's place in the electronic records chain of trust

4.3.2 IBM System Storage Archive Manager

IBM System Storage Archive Manager (SSAM) helps meet data retention and disposition regulations and policies:

- ▶ SSAM protects data by disallowing explicit data deletion, prior to the retention criteria.
- ▶ SSAM manages data by leveraging retention policies and expiration processes.
- ▶ SSAM offers choices about where to store data by exploiting the extensive device support of Tivoli Storage Manager.
- ▶ SSAM works with the Tivoli Storage Manager archive client, content manager, and archive applications to make data easily retrievable.

SSAM runs as and requires a separate Tivoli Storage Manager server instance that has the data retention option turned on during server setup. Note that multiple server instances can run in the same machine.

SSAM accepts data via the following interfaces:

- ▶ The Tivoli Storage Manager API
- ▶ The Tivoli Storage Manager archive client starting from Version 5.3.3

Content management and archive applications send data as an archive object to the Tivoli Storage Manager server via the Tivoli Storage Manager client application programming interface (API). No other data, such as backups, Space Management data, or data base backups, can be stored on this server instance.

You can use all of the robust device support Tivoli Storage Manager provides. And you can use all the powerful functions, for example, expiration, off-site copy creation, and collocation.

For additional documentation, see the SSAM page in the *IBM Tivoli Storage Manager Server Administration Guide* for your server platform, and refer to the following Web sites:

- ▶ *IBM Tivoli Storage Manager Using the Application Program Interface*, GC32-0793:
http://publib.boulder.ibm.com/tividd/td/TSMC/GC32-0793-03/en_US/PDF/ansa0000.pdf
- ▶ IBM Tivoli Storage Manager external Web site:
<http://www-306.ibm.com/software/tivoli/products/storage-mgr-data-reten/>

Setting up and using an SSAM server

The setup of an SSAM server is relatively simple. The installation procedure is the same as that of any Tivoli Storage Manager server. After installing the normal Tivoli Storage Manager server code, you have to keep in mind these major differences and requirements:

- ▶ You must have a license package consisting of the IBM Tivoli Storage Manager Extended Edition license, plus the SSAM license.
- ▶ You must have defined valid Tivoli Storage Manager policies.
- ▶ The Tivoli Storage Manager API on the client must be enabled for communication with a SSAM server by specifying the following option in the client system options file (`dsm.opt` in Windows or `dsm.sys` in UNIX):

```
enablearchiveretentionprotection yes
```
- ▶ You must have a dedicated SSAM server instance that is used only for data retention.
- ▶ The **set archiveretentionprotection** option must be set when preparing the server.
- ▶ You should have an enterprise content manager application or archive application, such as DB2 Content Manager, to send the data to the SSAM server via the Tivoli Storage Manager API or the Tivoli Storage Manager client.
- ▶ The SSAM server requires one or more storage pools to meet your performance and capacity requirements.

Attention: After archive retention protection is turned on, you cannot turn it off. There is no way of disabling this option as long as the server contains valid data. When the server contains no more valid data, there is little scope in turning off this option, because turning off the option would allow you to delete data, but there is no longer any data to delete.

Your interpretation of the regulations will dictate the choice of storage devices. SSAM can attach both WORM and normal rewritable media.

Starting with Version 5.3, data encryption using a 128-bit Advanced Encryption Standard (AES) is now available for the Archive API Client. Data can now be encrypted before transmission to the SSAM, so that it is then stored on the disk or tape in an encrypted format.

Table 4-1 summarizes the differences between SSAM and Tivoli Storage Manager Extended Edition.

Table 4-1 IBM Tivoli Storage Manager Extended Edition and SSAM

Function	IBM Tivoli Storage Manager Extended Edition	IBM System Storage Archive Manager
Install	IBM Tivoli Storage Manager Extended Edition CD	IBM Tivoli Storage Manager Extended Edition CD <i>and</i> set archiveretentionprotection
Devices supported	More than 400	More than 400
Server-to-server backup	Yes	No
Library sharing	Yes	Yes
Client data	Backup, archive, and Space Management	Archive from 5.3.3
API data	Backup and archive	Archive
Import/export data	Yes	No
Delete data, node, file space	Yes	No
Lower archive retention criterion	Yes	No
Archive hold/release	No	Yes
Chronological archive	Yes	Yes
Event-based archive	No	Yes

SSAM safety features

To ensure that objects stored under data retention policies remain compliant to those policies, the following restrictions apply with the use of Tivoli Storage Manager features:

- ▶ A registered node cannot be reassigned to a different policy domain.
- ▶ You cannot define a device class with device type SERVER. This means that you cannot use server to server virtual volumes to store data on another Tivoli Storage Manager server.
- ▶ You cannot import data to a Tivoli Storage Manager for a Data Retention server.
- ▶ You cannot activate a policy set that contains weaker retention parameters than the ones in place in the active policy set.
- ▶ You cannot remove data retention protection on a Tivoli Storage Manager for Data Retention server before the retention requirements for all data have been satisfied and all data has expired.

On SSAM servers with archive retention protection enabled, the following operations will not delete objects whose retention criteria have not been met:

- ▶ Requests from the client to delete an archive object
- ▶ DELETE FILESPACE from either a client or administrative command line
- ▶ DELETE VOLUME DISCARDATA=YES
- ▶ AUDIT VOLUME FIX=YES

Note: A cached copy of data can be deleted, but data in primary and copy storage pools can only be marked damaged and is never deleted until the data reaches its expiration date.

4.3.3 SSAM archive API options for data retention

Archive objects have new ways of being managed. They can use the standard chronological retention. After the object is sent to Tivoli Storage Manager, a clock starts a count down to the time when the object should expire.

A new event-based retention allows the count down clock to start after a specific event occurs. After a specific event occurs, the content manager program sends an event call via the API to Tivoli Storage Manager telling Tivoli Storage Manager to start the count down.

Archive copy group retention parameters

In order to use the archive function of Tivoli Storage Manager, you must define valid policies that preclude defining a policy domain, policy set, management class or classes, and an archive copy group, as well as setting archive retention parameters in the archive copy group and associating your application clients (applications using the API) with the Tivoli Storage Manager policies.

The archive copy group parameters that govern retention are RETVER, RETINIT, and RETMIN. The RETINIT and RETMIN parameters were introduced in Tivoli Storage Manager Version 5.2.2 to make it possible for applications using the API to further control the retention period (RETVER) for archive objects. Chronological archive retention has always been possible with Tivoli Storage Manager and was controlled solely by the RETVER parameter. With Tivoli Storage Manager V5.2.2, we have introduced event-based archive retention and two new archive copy group parameters.

Two methods of archive retention

There are two methods of archive retention, which are defined by the parameters of the archive copy group:

- ▶ Chronological archive retention
- ▶ Event-based archive retention

We now look at the parameters of the archive copy group and their possible values for the two archive retention methods.

The existing archive retention parameter

The existing archive retention parameter is RETVER (retain version). Possible values are RETVER=0 to 30,000 days or Nolimit.

The retain version parameter (RETVER) within the archive copy group specifies the number of days to retain each archive object. Possible values are 0 to 30,000 days or NOLIMIT, which means that an archive copy is maintained indefinitely.

New archive retention parameters

The two new archive retention parameters are RETINIT and RETMIN, which act as follows:

► **RETINIT (retention initiation):**

The possible values are RETINIT=creation or event.

The retention initiation (RETINIT) parameter specifies when the time specified by the retain version (RETVER=*n* days) attribute is initiated. The possible values for this parameter are creation or event. The default value is creation. In the following list, we explain both values:

- RETINIT=creation (chronological archive retention)

By setting this parameter to creation (RETINIT=creation) in the archive copy group, you specify that the retention time specified by the RETVER attribute (RETVER=*n* days) is initiated right at the time an archive copy is stored on the server. This is referred to as *chronological archive retention*.

- RETINIT=event (event-based archive retention)

By setting this parameter to event (RETINIT=event) in the archive copy group, you specify that the retention time (RETVER=*n* days) for the archived data is initiated by an application that used API function calls. If the application never initiates the retention, the data is retained indefinitely. This method of archive retention is referred to as *event-based archive retention*.

Possible events to signal through the API to the Tivoli Storage Manager server are:

- Activate: Activates the countdown of the RETVER value for the given object.
- Hold: Prevents the Tivoli Storage Manager server from deleting the object, even if the RETVER period has ended. Signaling a “hold” virtually does not extend the retention period, but a hold object will only expire after a release event is sent.
- Release: Removes the hold status of an object. The Tivoli Storage Manager server will then treat the object again according to the RETVER and RETMIN values.

► **RETMIN (retain minimum):**

Possible values are RETMIN=0 to 30,000 days.

The retain minimum (RETMIN) parameter applies only to event-based archive retention policy and specifies the minimum number of days to retain an archive object regardless of the value of RETVER. The default value is 365. Possible values are 0 to 30,000 days.

Data retention protection

Data retention protection ensures that archive objects will not be deleted from the Tivoli Storage Manager server until the policy-based retention requirements for that object have been satisfied. After an archive object is stored on a Tivoli Storage Manager for Data Retention server, retention protection cannot be removed. Retention protection is based on the retention criterion for each object, which is determined by the RETVER and RETMIN parameters of the archive copy group of the management class to which the object is bound.

If an object uses event-based retention (RETINIT=EVENT), the object will not expire until whatever comes later. Either the date the object was archived plus the number of days in the RETMIN parameter, or the date the event was signaled plus the number of days specified in the RETVER parameter. When using the chronological retention (RETINIT=CREATION), the archive object will expire after the time that is set with the RETVER parameter has elapsed.

Table 4-2 shows the relationship between the different parameters and their use within certain retention policies.

Table 4-2 Archive copy group parameters

Archive copy group parameters	Chronological retention	Event-based retention
RETINIT Defines when to initiate retention period defined in RETVER attribute.	RETINIT=CREATION Expiration date is based on the date the object was archived plus RETVER.	RETINIT=EVENT Expiration date is based on date of retention initiation event plus RETVER.
RETVR Number of days to retain the archive object after retention is initiated.	RETVR=0 to 30,000 days or NOLIMIT	RETVR=0 to 30,000 days
RETMIN Minimum number of days to retain archive object.	Not applicable.	RETMIN=days Based on date object was archived.
What is the earliest date that the object could become eligible for expiration after retention has been initiated?	(date object was archived) + RETVER	(date retention was initiated through Event) + RETVER or (date object archived) + RETMIN Whichever is longer.

Chronological archive retention

Figure 4-17 shows a simplified view of a chronological retention policy. With settings of RETINIT=creation and RETVER=365 days, a file that is archived on day 0 is retained for 365 days and becomes eligible for expiration. In this case, after 365 days from the time the data was created, all references to that data are deleted from the database, making the data irretrievable from Tivoli Storage Manager storage volumes. This kind of archive retention is called *chronological retention*. By default, the RETINIT value is set to creation.

Note: Choose chronological archive retention when the application that is doing the archiving is not able to send retention events such as activate, hold, and release. Also use chronological archive retention when you archive to a regular Tivoli Storage Manager server (not enabled for data retention protection) through the normal backup-archive client.

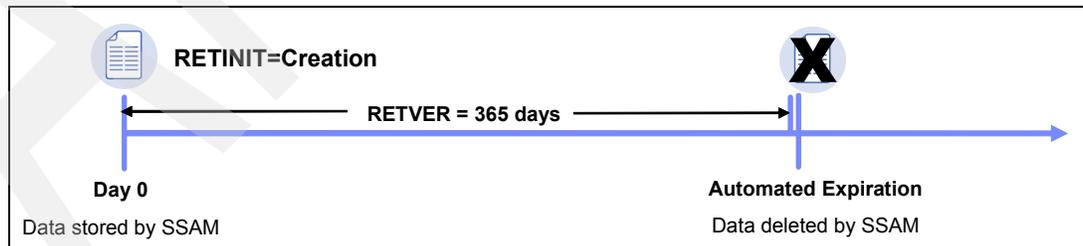


Figure 4-17 Chronological retention policy

Archive copy groups using a chronological retention policy satisfy many archive retention requirements.

Event-based retention policy

In certain situations, it is hard to define data retention periods, or they depend on events taking place long after the data is archived. Event-based archive retention is designed to meet these requirements. The event-based retention policy is designed for applications that use the API function calls to trigger events also known as *retention events*.

Figure 4-18 shows a timeline depicting event-based policy. In this example, an application using the API archives data using the retention values is shown. The archived data is retained for a minimum of 2,555 days (RETMIN=2555). If the retention time (RETVER) is activated through an API retention event, Tivoli Storage Manager assigns an expiration date for this object. The expiration date that Tivoli Storage Manager assigns is whichever comes later, either one or the other:

- ▶ The date the object was archived, plus the number of days specified in the RETMIN parameter.
- ▶ The date the event was signaled, plus the number of days specified in the RETVER parameter.

After reaching this expiration date, the data is eligible for expiration. When the time for expiration occurs, all references to that data are deleted from the Tivoli Storage Manager database, making the data irretrievable from Tivoli Storage Manager storage volumes. This kind of archive retention is referred to as event-based retention.

Note: Use event-based archive retention if the archive application you are using (such as DB2 Content Manager together with Record Manager) uses the API function calls to control the retention period of the archived data objects.

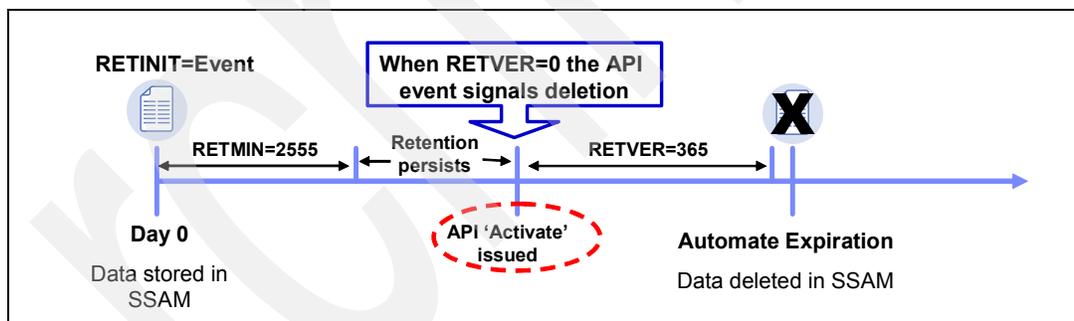


Figure 4-18 Event-based retention policy

Deletion hold and release

Some regulations require that you retain data longer than the minimum retention period in certain cases. This might be due to any litigation, a legally-required or a company-required audit, or criminal investigation requiring the data as evidence. The API supports new function calls used to place a deletion hold on an archive object. These functions are also called retention events.

A deletion hold can be applied at any point in time during the retention period for an archive object. The object will then be retained until a deletion release is applied. If a deletion release is not applied, the object is retained indefinitely. Although deletion hold and release are events, they can be applied to objects archived not only using the event-based policies, but also the chronological, creation-based policies.

Figure 4-19 shows a timeline depicting deletion hold and release for an object stored with a chronological retention policy.

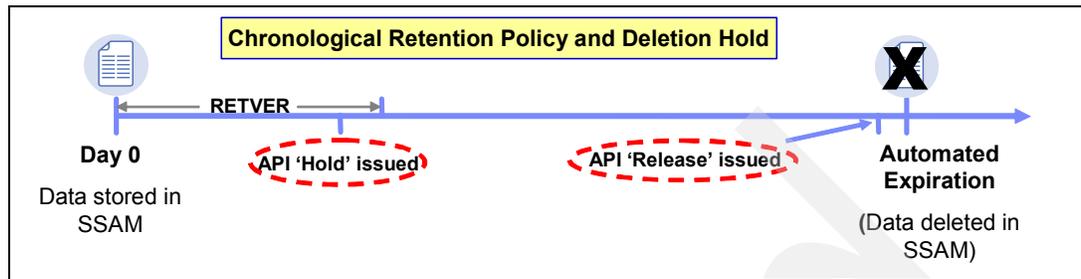


Figure 4-19 Chronological retention policy deletion hold and release

Figure 4-20 shows a timeline depicting deletion hold and release for an object stored with an event-based retention policy. Note that the API hold is issued after the RETMIN period. The object has not yet expired when the API hold is issued because RETINIT=event and no event has been issued.

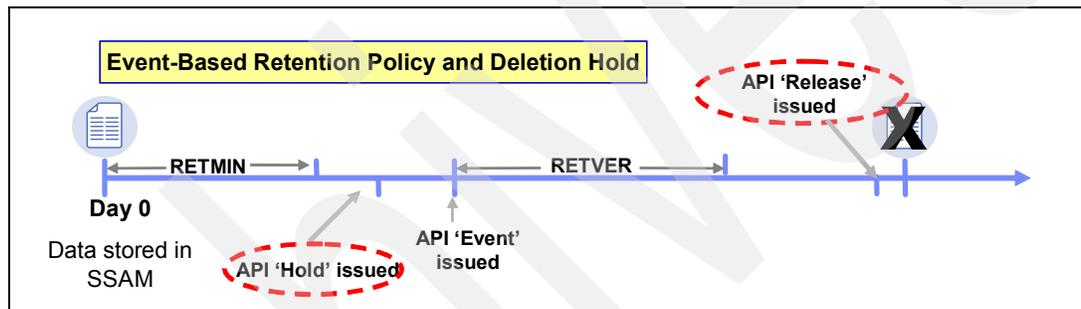


Figure 4-20 Event-based retention policy deletion hold and release

4.3.4 Storage hardware options for Archive Manager

SSAM supports more than 400 storage devices. These are the same devices that Tivoli Storage Manager Extended Edition supports. Depending on the regulatory requirement that customers are trying to meet, there might or might not be specific types of media required.

Most regulations allow the stored data to be on any type of device as long as the content management application establishes a retention policy. This capability is now changing. For example, in many cases the old paradigm was to have regulatory data stored on optical media, and now the ability has opened up to store data on other types of media, such as disk and tape.

Tip: IBM recommends using the IBM TotalStorage 3592 Enterprise Tape Drive in combination with the IBM TotalStorage 3592 WORM media, or the new generation of IBM Ultrium 3 LTO drives in combination with the 3589 WORM media, to complement the SSAM characteristics for non-erasable and non-rewritable data on the tape storage pool.

For more information about WORM media support, see Chapter 8 and the heading titled “Special Considerations for WORM Tape Media” in the *IBM Tivoli Storage Manager for AIX Administrator's Guide Version 5.3*, GC32-0768.

4.4 IBM System Storage N series SnapLock feature

The IBM N series SnapLock function is a data function designed to deliver high performance and high-security disk-based file locking or WORM functionality on both near-line and primary IBM System Storage N series storage. The SnapLock function can help manage the permanence, accuracy, integrity, and security of data by storing business records in an inalterable form and allowing for their rapid online accessibility for long periods of time. There are two versions of SnapLock:

- ▶ SnapLock Compliance: For strict regulatory environments
- ▶ SnapLock Enterprise: For environments without regulatory restrictions

4.4.1 SnapLock Compliance

SnapLock Compliance is designed to help organizations address strict records retention regulations. Protection is offered on two levels:

- ▶ Users or administrators are prevented from deleting or modifying individual SnapLock Enterprise WORM records until the records have expired.
- ▶ Administrators are prevented from deleting SnapLock Enterprise volumes that contain the WORM records until all records on the volume have expired.

4.4.2 SnapLock Enterprise

SnapLock Enterprise supports adherence to rigorous organizational best practices through functionality similar to that of SnapLock Compliance, but allows administrators to delete entire SnapLock Enterprise volumes.

N series stores data in volumes and these volumes contain files. The files are stored on these volumes by applications, in our case data archival applications. To use SnapLock, you must create a SnapLock volume. Files are then archived to the SnapLock volume by writing them using the CIFS or NFS file sharing protocols.

After you place a file into a SnapLock volume, you must explicitly commit the file to a WORM state. This is done by setting the *last accessed* time stamp to the desired retention date and then making the file become read only. After the file is committed to the WORM state, no alterations, overwrites, or deletions are possible until file expiration. Files not explicitly committed to the WORM state are protected by an administrator-defined minimum retention period.

Data can be appended to a SnapLock file before a file is committed. This means that the file can be closed and subsequently reopened multiple times. After the file is committed and set to read only, SnapLock will not allow any subsequent appends to the file.

Figure 4-21 illustrates the use of SnapLock with an archival application such as Tivoli Storage Manager.

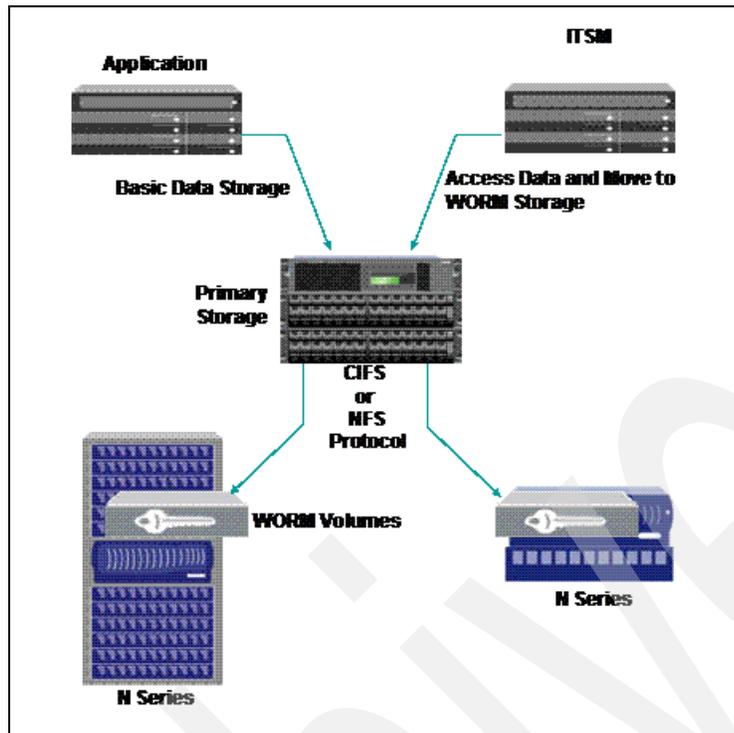


Figure 4-21 N series SnapLock data flow

4.4.3 SSAM and IBM N series

In this section, we discuss how SSAM can take advantage of IBM N series devices with the SnapLock feature and explain what to consider when implementing this solution.

The SnapLock feature allows for applications such as SSAM to set a retention date for a file and commit the file to the WORM state. Using this feature, SSAM is responsible for the protection of the metadata by not allowing operations such as accidental or intentional deletion of data from the SSAM server while the SnapLock feature protects the physical data on the SSAM storage volume from accidental or intentional deletion.

You can only use the SnapLock feature with Tivoli Storage Manager servers that have the Data Retention Protection (DRP) feature enabled: SSAM and DR550. The SnapLock feature is not used by a standard, non-DRP protected Tivoli Storage Manager server. The SnapLock support requires Tivoli Storage Manager Version 5.3 or later.

How SSAM stores data into IBM N series

Data archived to a SSAM server and stored on IBM N series system storage is stored as a Tivoli Storage Manager file volume. This is a Tivoli Storage Manager volume mapped to a Tivoli Storage Manager file device class. The Tivoli Storage Manager file device class represents a collection of files on a file system, where the file system can be locally attached or network attached as in the case of an N series file system. Figure 4-22 illustrates the interaction of SSAM and N series.

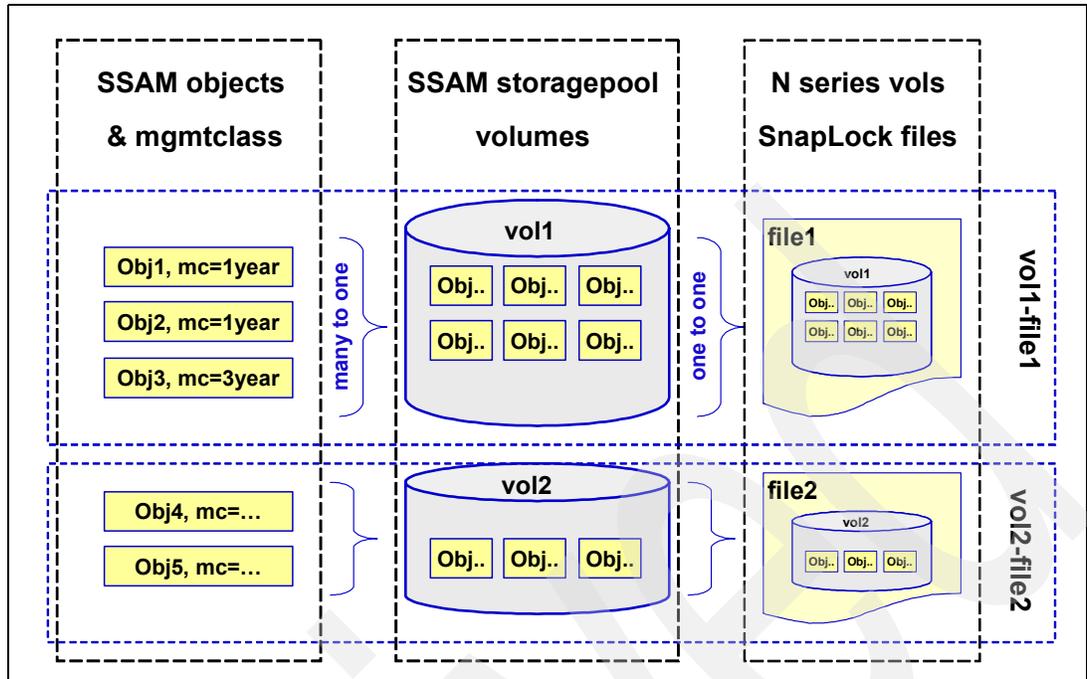


Figure 4-22 Interaction between SSAM and IBM N series objects

Objects are stored in the SSAM server by a storage management application. When an object is stored in SSAM, the application requests a unique management class out of many possible predefined management classes. Each individual management class contains two important pieces of information:

- ▶ Retention: How long to keep the object
- ▶ Destination: Where to put (store) the object, also known as Tivoli Storage Manager destination storage pool

For the scope of our discussion, we assume that the destination storage pool is mapped to a device class that points to a N series storage system.

After the object is assigned a unique management class, this will determine where the data is located and how long it will be stored. In the example shown in Figure 4-22, we see three separate objects, *obj1* to *obj3*, with different retentions that are stored in one SSAM storage pool volume, *vol1*. Then we see two more objects, *obj4* and *obj5*, stored on a different volume, *vol2*. We have a *many-to-one* relationship between archived objects and SSAM volumes, because multiple objects can be stored on one individual volume. Different archive objects on the same volume object can have different retention dates, based on the management class assigned to that object. The retention of the SSAM volume is set to the longest retention period or highest retention of the different objects.

SSAM volumes are stored as individual files on the N series storage system, there is a one to one relationship between a SSAM volume and an N series storage system, therefore, as the example in Figure 4-22 illustrates, SSAM *vol1* corresponds to N series *file1*, *vol2* to *file2*, and so on. The retention of the N series volume is determined by Tivoli Storage Manager as that of the object with the highest retention, in our case we have management classes with retention of one and two years and the retention of volume *file1* in our example will be set to two years.

4.4.4 IBM N series tiered storage

N series storage systems can support multiple tiers of disk storage devices in the same filer — for example, you can mix fast Fibre Channel disk with lower cost SATA disk drives.

To understand how N series storage and SSAM storage interact, it is necessary to introduce some basic N series storage concepts, because N series has an integrated Logical Volume Manager (LVM) function:

- ▶ The *disks* represent the physical hardware, real disk drives. Each disk has a name based on the physical address of the disk drive.
- ▶ The *aggregates* are named collections of *disks* that are managed as a group and have the same raid and parity properties. Aggregates can be extended by adding disks.
- ▶ The *volumes* are named entities of storage that are accessed by clients as network file systems using the CIFS or NFS protocol. One *aggregate* can contain multiple *volumes*.

Tip: In the context of this book, the term *volume* refers to different entities in different contexts. An IBM N series volume is used by a host client by mounting it as a network file system through the CIFS or NFS protocol. A SSAM or Tivoli Storage Manager volume is a Tivoli Storage Manager storage entity that uniquely represents a removable volume, such as a tape cartridge or a file on disk.

Tivoli Storage Manager volumes can be mapped to files on an IBM N series storage systems and these files are contained in one IBM N series volume. The IBM N series volume is mapped to a specific group of N series disks through aggregates.

You could choose to manage different sets of data in different ways, for example, by storing data that frequently must be accessed on high performance IBM N series disk and data that require less frequent access on cheaper ATA storage devices. Figure 4-23 shows the interaction of SSAM and IBM N series to obtain a tiered storage environment.

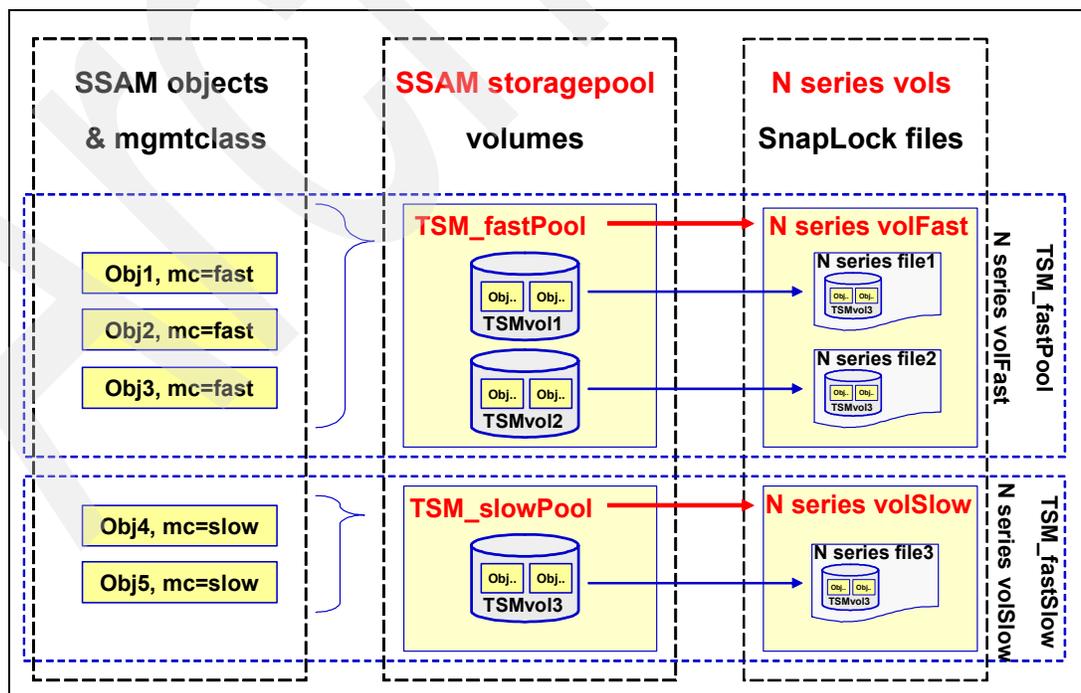


Figure 4-23 SSAM use of N series tiered storage

In the example in Figure 4-23, we have defined two different IBM N series volumes:

- ▶ *volFast* is defined on high performance SCSI disk.
- ▶ *volSlow* is defined on ATA storage devices.

We have then defined two separate Tivoli Storage Manager storage pools with a device class of file and a directory parameter indicating the mount point of the shared network file system exported by the N series storage system. *TSM_fastPool* is mapped to a directory corresponding to N series *volFast* while *TSM_slowPool* is mapped to a directory corresponding to *volSlow*. In this manner, Tivoli Storage Manager volumes that are created in the *TSM_fastPool* storage pool are stored in the IBM N series *volFast*, and, in the same manner, volumes created in the *TSM_slowPool* are stored on *volSlow*.

SSAM data is stored in the different storage pools through the Tivoli Storage Manager management class construct: In the example, objects *obj1* to *obj3* have a management class of fast that is configured to point to the *TSM_fastPool* while objects *obj4* and *obj5* have a management class that points to the *TSM_slowPool* storage pool.

SSAM reclamation and SnapLock

In “How SSAM stores data into IBM N series” on page 104, we discussed how SSAM stores data into N series filers with the SnapLock feature. SSAM management class policies determine and manage the retention period for WORM file volumes. The SSAM retention date is calculated by determining the greatest value of the SSAM RETVER and RETMIN retention parameters of all files that are stored on a specific retention-managed volume and adding one month. The volume can be filled in multiple Tivoli Storage Manager transactions or client sessions. On each transaction, the greatest retention value is determined and written to the SSAM volume/IBM N series file as the last reference date. After the volume is filled to a SSAM administrator-defined volume maximum capacity, the volume is committed to WORM state and the N series file last reference date is set to the calculated SSAM retention date.

In some cases, the retention of individual files can exceed the retention date of the SSAM volume on which the files were originally stored. Some objects on the volume might have to be retained longer than other objects on the same volume, because of various reasons:

- ▶ They are bound to management classes with different retention times.
- ▶ They are managed by event-based retention and the event has not yet occurred.
- ▶ They cannot be removed because of a deletion hold.
- ▶ The retention for a copy group might be increased, requiring a longer retention time than that specified to SnapLock when the WORM FILE volume was committed.

In these cases, the valid files must be transferred to another volume before the original volume expires to ensure that they are retained on WORM media. SSAM is instructed to perform this kind of management when the SSAM storage pool reclamation type parameter is set to RECLAMATIONTYPE=SNAPLOCK. This parameter applies only to individual SSAM server file storage pools that point only to N series volumes with SnapLock feature enabled.

There are three retention periods available at the individual N series volume level with the SnapLock feature and it is important to configure them correctly for interaction with SSAM retention. Separate N series volumes can have different retention periods based on the data that is stored in them. These retention periods are:

- ▶ Minimum retention period: It defines the shortest amount of time that a file will be retained in an N series filer. By default it is set to 0. The recommendation is to set it to the highest of the following two values:
 - The minimum value, which is 30 days
 - The minimum retention indicated in any Tivoli Storage Manager copy group pointing to the N series volume

- ▶ **Maximum retention period:** If the maximum data retention is less than 30 years, we suggest that you leave the default, 30 years. This allows Tivoli Storage Manager to control the retention period.
- ▶ **Default retention period:** Used if the application fails to assign a retention period. We suggest that you use the default, 30 days.

Setting the retention periods according to these rules will ensure that SSAM can manage SnapLock storage pools with the maximum efficiency.

Tip: When using Tivoli Storage Manager event based retention, stored data does not have an expiration date assigned. You should set the maximum retention period for the storage pool to the average life expectancy of the data. This forces a reclamation to occur after that period, to free the space used by any expired data.

For each volume in an SSAM storage pool volume, a reclaim period is created; the reclaim period is defined by the SSAM volume `BEGIN_RECLAIM_PERIOD` and `END_RECLAIM_PERIOD` attributes. The reclaim period is defined as a time period that starts a number of days prior to when the SSAM volume retention date is to expire and ends when the SSAM volume expires.

During the reclaim period, any unexpired objects remaining on the SSAM volume will be copied to another SSAM volume. The `BEGIN_RECLAIM_PERIOD` is defined as the greatest expiration date of all objects on the volume, while the `END_RECLAIM_PERIOD` is defined as the `BEGIN_RECLAIM_PERIOD` plus one month. The `END_RECLAIM_PERIOD` is also used as the retention of the file in the N series filer.

This means that the volume will be retained in the IBM N series storage system for approximately one month after the `BEGIN_RECLAIM_PERIOD` value, after this it will be automatically expired by the IBM N series storage system based on its retention date. The reclaim period allows SSAM to move any valid data on an existing volume to new storage volumes before the original volume is expired. During the volume's reclaim period, Tivoli Storage Manager will automatically move any valid data objects to new SnapLock protected storage pool volume.

In Figure 4-24, we illustrate the steps in the life of a SSAM volume and the related N series SnapLock file.

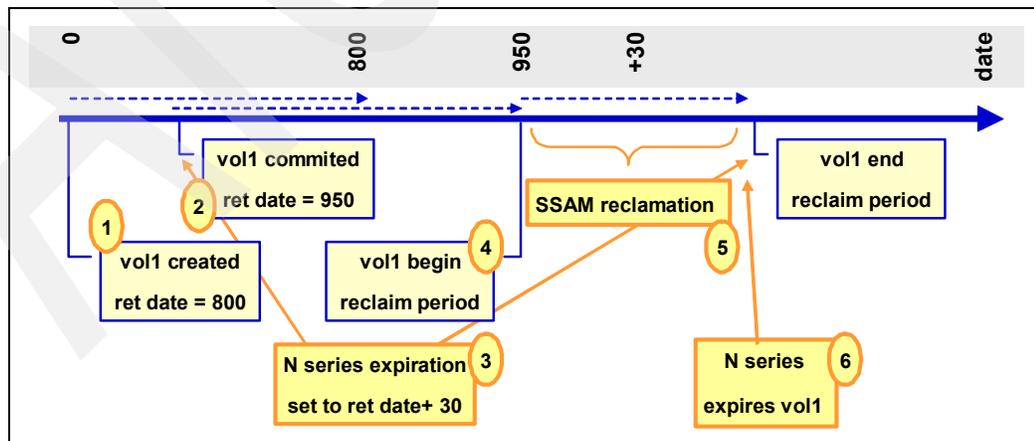


Figure 4-24 SSAM and IBM N series volume expiration

The date axis shows an arbitrary timeline starting at zero. Let us see how the reclaim periods are calculated:

In step 1, the volume *vol1* is created and the greatest retention date of all objects on it is determined to be 800 days.

4. In step 2, more data is stored to the volume and the retention date is recalculated because there are files that will expire in 950 days. The volume fills to maximum capacity and is closed and the data committed to IBM N series SnapLock.
5. The IBM N series expiration date is calculated as the maximum SSAM retention date plus 30 days as shown in 3.
6. In step 4, the reclamation period starts for *vol1*. SSAM will allow some latitude for expiration processing to expire most of the data on the volume in order to minimize the amount of data to move.
7. In step 5, we show that SSAM has a one month window in which to perform volume reclamation and move data to new SnapLock volumes.
8. In step 6, at the end of the reclamation period for *vol1*, Tivoli Storage Manager reclamation processing will check for empty SnapLock file volumes whose retention dates have expired and delete them from the IBM N series storage system.

Important: We do not recommend disabling reclamation processing on storage pool volumes with the SnapLock feature enabled, because SSAM cannot issue warning messages that data will become unprotected and cannot move the data to new volumes before the original one expires.

Archived

Tiers of storage

In this chapter we describe the different data storage products from IBM System Storage. They can be utilized to build a tiered storage environment to support an Information Lifecycle Management (ILM) solution. IBM offers several other storage products, for instance, through the server brands. These offerings can be, like any other storage solution, a valid storage tier. The reason why they are not described here is because it is not our objective in this book to give a complete overview of all available storage products. However, the products in this chapter can be seen as the most common components for a tiered storage solution.

We cover the following products:

- ▶ Disk storage:
 - DS8000 series,
 - DS6000 series,
 - DS4000 series,
 - N series
- ▶ Optical storage:
 - IBM 3996
- ▶ Tape storage:
 - LTO Ultrium,
 - IBM 3592 and TS1120
- ▶ Virtualization solutions:
 - SAN Volume Controller
 - TS7510
 - IBM 3494 Virtual Tape Server

For a comprehensive overview of all IBM System Storage and TotalStorage offerings, refer to:

- ▶ The IBM System Storage Web site:
<http://www.storage.ibm.com>
- ▶ *IBM System Storage Solutions Handbook*, SG24-5250, which is available for download at:
<http://www.redbooks.ibm.com/Redbooks.nsf/RedbookAbstracts/sg245250.html?Open>

5.1 Storage tiers

As described in 2.1.2, “The fluctuating value of data” on page 30, data has different value over time and by data type. According to the Storage Networking Industry Association (SNIA) and analyst studies, the Total Cost of Ownership (TCO) for storing data is much cheaper for a mixed disk/tape environment versus disk only. It costs less than disk, yet is more responsive and flexible than tape. And the advantages of utilizing tiered storage go beyond just TCO:

- ▶ It matches the value of data to the cost of media.
- ▶ It creates an architecture to manage the coming explosive growth in data having to be archived in the future.
- ▶ It provides more automation, with less costly manual intervention.

Figure 5-1 shows the different tiers of disk and tape storage IBM can provide with the different server platforms.

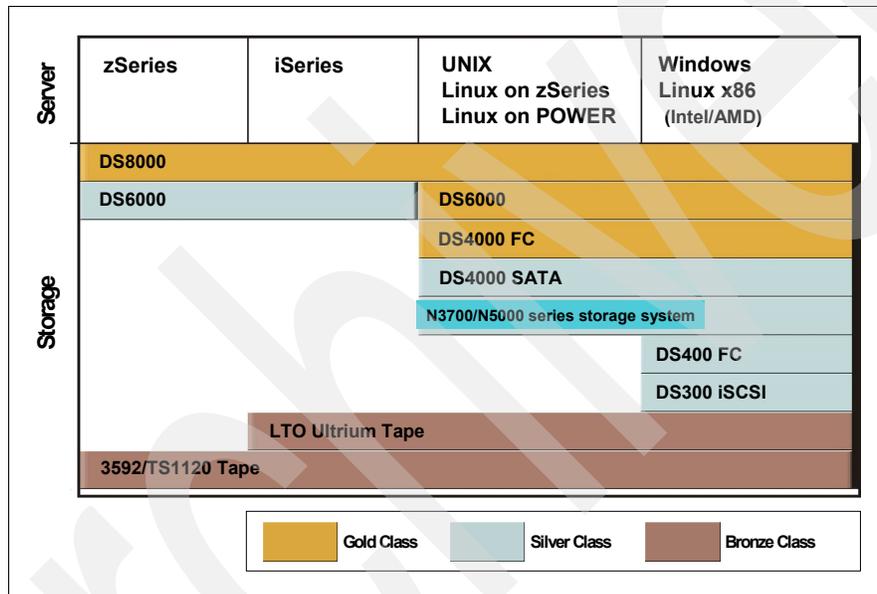


Figure 5-1 IBM disk and tape storage tiers for each server platform

Obviously every organization can set storage classes differently according to their requirements. It is important to understand that different storage solutions can have different purchase and operational costs. This difference can be utilized by an ILM solution to implement cost differentiation for storing data.

5.2 Enterprise disk systems

The IBM Enterprise Disk Systems are designed to deliver high-performance, high-availability storage with flexible characteristics that can be configured according to the business requirements. Building on a solid foundation of the IBM TotalStorage Enterprise Storage Server® (ESS) environment and reusing IBM technology innovations, the IBM TotalStorage DS8000 series, along with the D6000 series, delivers an enterprise storage continuum of systems with the same functional code, shared replication services, and common management interfaces.

Enterprise Storage Server (ESS)

The ESS set a new standard for storage servers back in 1999 when it was first made available. From the initial E models to the succeeding F models, to the current 750 and 800 models, the ESS significantly improved over time levels of performance, throughput, and scalability with more powerful hardware and functional enhancements.

The DS6000 series

The DS6000 series offers true enterprise-class functionality with modular design and reduced price. Clients who currently have IBM TotalStorage ESS models in their enterprise should also consider the IBM TotalStorage DS6000 series when they plan to replace or buy additional storage. Intended for medium and large businesses, the DS6000 series can help simplify data management and enable easy scalability.

The IBM DS6000 series offers IBM server iSeries™ and zSeries® customers for the first time the option for a mid-range priced storage subsystem with all the features and functions of an enterprise storage subsystem.

The maximum storage capability of the DS6800 controller is 4.8 TB. With the optional DS6000 expansion enclosures, a maximum storage capability of 64 TB can be reached.

The DS8000 Series

The IBM TotalStorage DS8000 series is the next generation of the IBM TotalStorage Enterprise Storage Server (ESS) designed for the most demanding, mission critical environments requiring the highest level of availability. The DS8000 series is designed to set an entirely new industry standard for high-performance, high-capacity by delivering unprecedented performance and scalability.

The physical storage capacity of the DS8000 series systems can range from 1.1 TB to 320 TB and it has an architecture designed to scale up to a petabyte. The DS8000 series allows additions and upgrades from one model to another to adapt to changing business requirements.

The DS6000/DS8000 series enables you to construct a multi-tiered storage environment to help minimize storage costs by retaining frequently accessed or high-value data on higher performance DS8000 storage servers and archiving less frequently accessed or less valuable information about less-costly DS6000 systems.

In the remainder of this section we describe the key common characteristics of the enterprise-class IBM disk storage products.

5.2.1 Storage consolidation

Consolidation begins with compatibility. The IBM Enterprise Disk Systems can be connected across a broad range of server environments. You can easily split up storage capacity among the attached environments and reduce the number of storage systems you have to use. At the same time, you can construct a disaster recovery solution that makes use of the full range of your Enterprise disk storage. For example, you can mirror a DS8000 series system with a DS6000 series system or an ESS.

5.2.2 Performance

The IBM Enterprise Disk Systems are designed for high performance that takes advantage of IBM leading technologies. In today's world, enterprises require business solutions that can deliver high levels of performance continuously every day, day after day. They also require a

solution that can handle different workloads simultaneously, therefore, they can run business intelligence models, large databases for enterprise resource planning (ERP), and online and Internet transactions alongside each other. Some of the unique features that contribute to the overall high-performance design of the IBM Enterprise Disk Systems are as follows.

Server-based design

The design decision to use processor memory as I/O cache is a key element of the IBM storage architecture. Performance improvements can be traced to the capabilities of the processor speeds, the L1/L2 cache sizes and speeds, the memory bandwidth and response time, and the PCI bus performance.

With the DS6000 (see Figure 5-2) and DS8000 series, the cache access has been accelerated further by making the non-volatile storage (NVS) a part of the main memory. Some part of the memory is used for the operating system and another part in each controller card acts as non-volatile storage (NVS), but most of the memory is used as cache. This design to use processor memory makes cache accesses very fast.



Figure 5-2 DS6000

IBM multipathing software

IBM Multipath Subsystem Device Driver (SSD) provides load balancing and enhanced data availability in configurations with more than one I/O path between the host server and the storage server. Most vendors' priced multipathing software selects the preferred path at the time of initial request. IBM free of charge preferred path multipathing software dynamically selects the most efficient and optimum path to use at each data interchange during read and write operations. The cost of vendor multipath software should be considered in the Total Cost of Ownership when comparing to the IBM DS6000 and DS8000.

Performance for zSeries

As is the case for the IBM TotalStorage ESS, the new DS6000 and DS8000 also supports the following IBM performance innovations for IBM Eserver zSeries environments.

Parallel Access Volumes (PAV)

PAV is an optional feature for zSeries environments, which enables a single zSeries server to simultaneously process multiple I/O operations that can help to significantly improve throughput. With Dynamic PAV, storage volumes can be automatically managed to help the workload meet its performance objectives and reduce overall queuing.

Multiple Allegiance

Multiple Allegiance is a standard feature which expands simultaneous I/O access capability across multiple zSeries servers. This function, along with the software function PAV, enables storage systems to process more I/O operations in parallel, helping to dramatically improve performance and enabling greater use of large volumes.

Priority I/O Queuing

Priority I/O Queuing improves performance in z/OS environments with several z/OS images. The z/OS Workload Manager (WLM) controls where work is run and optimizes the throughput and performance of the total system. The IBM TotalStorage Enterprise Disk Systems provide the WLM with more sophisticated ways to control the processing sequence of I/O operations.

FICON

The 2 GB FICON® connectivity delivers high bandwidth and provides a high-speed pipe supporting multiplexed operations for zSeries systems. The ESS and the DS8000 series provide ESCON® connection for older zSeries hosts that do not support FICON.

5.2.3 Data protection

Many design characteristics and advanced functions of the IBM Enterprise Disk Systems contribute to protect the data in an effective manner.

Fault-tolerant design

The IBM TotalStorage ESS and the DS8000 series are designed with no single point of failure. It is a fault-tolerant storage subsystem, which can be maintained and upgraded concurrently with user operations. The DS6000 series is also designed and implemented with component redundancy to help reduce and avoid many potential single points of failure.

RAID protected storage

The IBM TotalStorage Enterprise Disk Systems support RAID-5, RAID-10 configurations, or a combination of both. This gives you more flexibility when selecting the redundancy technique for data protection.

5.2.4 Common set of functions

The DS6000 series, the DS8000 series, and even the ESS storage subsystems share a common set of advanced functions, including FlashCopy®, Metro Mirror, Global Copy, and Global Mirror. Therefore, there is only one set of skills necessary to manage the whole enterprise disk storage systems.

There is also a set of common functions for storage management, including the IBM TotalStorage DS Command-Line Interface (DS CLI) and the IBM TotalStorage DS open application programming interface (API).

For more information about DS6000 and DS8000 series, refer to the Web page:

<http://www.storage.ibm.com/disk/enterprise>

5.3 Midrange disk systems

The DS4700 Express storage system is designed to address many of the requirements our customers have come to expect from the DS4000 Series disk storage products. Two models are available; the model 70 has 4 total host ports, 2 GB of cache, and high performance; the model 72 has 8 total host ports, 4 GB of cache, designed to provide the right processing power when required. Unless otherwise stated for differentiation, we will continue to refer to both models as “DS4700 Express” for simplicity. Figure 5-3 shows the DS4700.

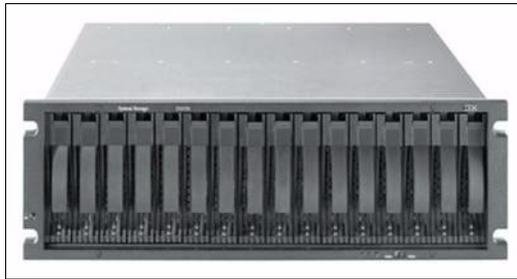


Figure 5-3 IBM System Storage DS4700 Express

The DS4700 Express storage system integrates 4 Gb/s Fibre Channel (FC) technology, designed for high-performance FC disk drives, integrated XOR engines, and powerful storage management functionality to help create, robust, high performance solutions targeted squarely at the midrange.

Designed specially for open systems environments, the DS4700 Express storage system's high-speed disk performance enables fast, responsive applications that can help improve transaction rates and customer satisfaction. Its modular “pay-as-you-grow” scalability can help lower acquisition and expansion costs by avoiding over-configuration and enabling optimal just-in-time purchasing. And with online scalability up to 33.6TB of Fibre Channel disk storage with attachment of six EXP810s, the DS4700 Express storage system easily satisfies demanding capacity requirements. Its 4 Gb/s host-side connectivity supports direct attachment to hosts (DAS) or storage area networks (SANs).

The DS4700 Express storage system's high availability helps keep data accessible and can help decrease the risk of downtime-related revenue loss. And its extensive compatibility results are designed to have minimal or no impact on existing infrastructure, helping to provide infrastructure investment protection. Figure 5-4 shows an EXP810.



Figure 5-4 DS4000 EXP810 expansion module

Performance

Compared to the DS4300, the DS4700 Express architecture combines faster processors, faster buses, more cache, an integrated drive-side loop switch and 4 Gb Fibre Channel technology to create a system that is designed to excel at both IOPS and bandwidth (MB/s). While certain competitive products might be promoted as 4 Gb, they might only be referring to their host interface, and do nothing to enhance the back-end (drive side) performance.

High Density Controller Enclosure

The 4 Gb enhanced controller enclosure of the DS4700 Express is designed for higher capacity compared to DS4300. The DS4700 Express is an integrated 3U chassis including two controllers, dual power, cooling and battery back-up units and up to sixteen (16) 3.5 inch hot-pluggable disk drives. Fibre Channel and SATA disk drives are both supported, as well as mixing those two technologies within the same enclosure. Up to six additional disk drive expansion units, such as the DS4000 EXP810, can be attached to the DS4700 Express for a maximum total of 112 disk drives.

Compatibility

Designed to help extend the backward and forward compatibility of the DS4700 Express. Host ports, for example, were designed with speed auto-negotiation logic, enabling connection to 1, 2, or 4 Gb host interfaces, in consideration of possible installed legacy hardware. DS4000 EXP710 2 Gb disk drive expansion enclosure can be attached to DS4700 Express, with or without the inclusion of the DS4000 EXP810 4 Gb disk drive expansion enclosure.

Connectivity

With 8 host ports for attachment of either hosts or switches, and 4 drive side loops, there is double the connectivity and the potential for higher performance when compared with earlier products.

Configurations

The DS4700 Express Storage™ System is offered in two models, 72 and 70. Because of the high level of integration of this product line, the models might appear quite similar, but can be differentiated in terms of connectivity, standard storage partitions and cache size. Table 5-1 shows the differences between the model 72 and model 70 of the DS4700 Express.

Table 5-1 DS4700 Express model differences

DS4700 Express Model 72	DS4700 Express Model 70
Eight 4 Gb/s host ports	Four 4 Gb/s host ports
Four 4 Gb/s drive ports	Four 4 Gb/s drive ports
4 GB of controller cache	2 GB of controller cache
Integrated XOR engine	Integrated XOR engine
“High” performance	“High” performance
16 integrated disk drive slots	16 integrated disk drive slots
Max of 112 drives (6 additional drive enclosures)	Max of 112 drives (6 additional drive enclosures)
2/4 Gb/s FC drives (mixed FC/SATA II previewed)	2/4 Gb/s FC drives (mixed FC/SATA II previewed)
DS4000 Storage Manager software	DS4000 Storage Manager software
Partitions, min. 8, max. 64	Partitions: min. 2, max. 64
FlashCopy	FlashCopy
Volume Copy	Volume Copy
Enhanced Remote Mirroring	Enhanced Remote Mirroring

The DS4700 Express supports 4 Gb DS4000 EXP810 and 2 Gb DS4000 EXP710 disk drive Expansion Enclosures behind the same DS4700 Express; however, its drive side loops must run at the same speed.

One of the benefits of the DS4700 Express is the ability to intermix 2 Gb and 4 Gb FC disk drives within the same DS4700 Express controller. As is the case when mixing drive modules. However, mixing 2 Gb and 4 Gb FC drives will require the entire 4 Gb/s enclosure to run at 2 Gb/s speed. This includes its internal loops to the drives and external drive loop interfaces. The link speed is set by a switch on the front of the enclosure.

Technical overview

The designs prevalent throughout the DS4700 Express begin with the hardware enclosure packaging. All current enclosures employ the same 3U rack mount chassis. Refer to the block diagram in Figure 5-5.

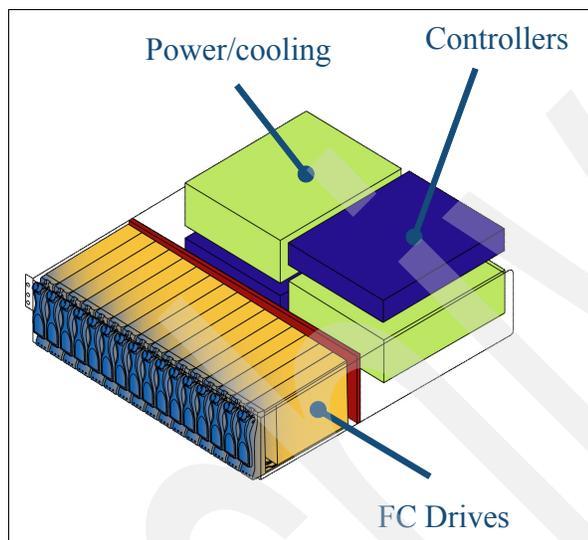


Figure 5-5 Basic building blocks DS4000 modules, including DS4700 Express

The primary chassis in the DS4700 Express Storage System is the Controller Module. As shown, this rack mount unit has capacity for 16 Fibre Channel Enhanced Disk Drive Modules (E-DDMs). The E-DDMs are easily removable and replaceable, hot, plugging into a proprietary midplane, which is field replaceable also. In the controller module, the drives are recessed behind a functional, decorative bezel.

As with the front, the rear of the DS4700 Express is also fully accessible for cabling. In Figure 5-5, the controller housing (shown in the deep blue) is mounted adjacent to its companion dedicated power and cooling unit (shown in green). The hot replaceable cache backup battery unit connects separately to the controller.

Figure 5-6 shows the components visible from the rear of the DS4700 Express Controller Module.

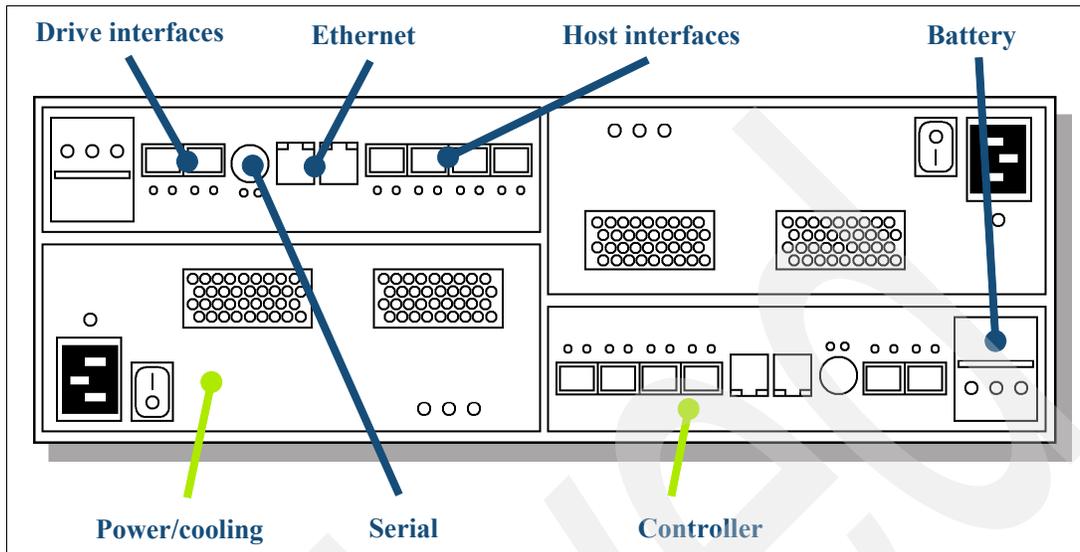


Figure 5-6 DS4700 Express Controller Module rear view

Figure 5-6 shows a DS4700 Express Model 72, because each of the dual controllers has 4 host interfaces. The equivalent DS4700 Express Model 70 visual would look identical, with the exception that the each controller would have only 2 host ports.

Figure 5-7 is a view of the rear of the DS4000 EXP810 4 Gb Expansion Enclosure.

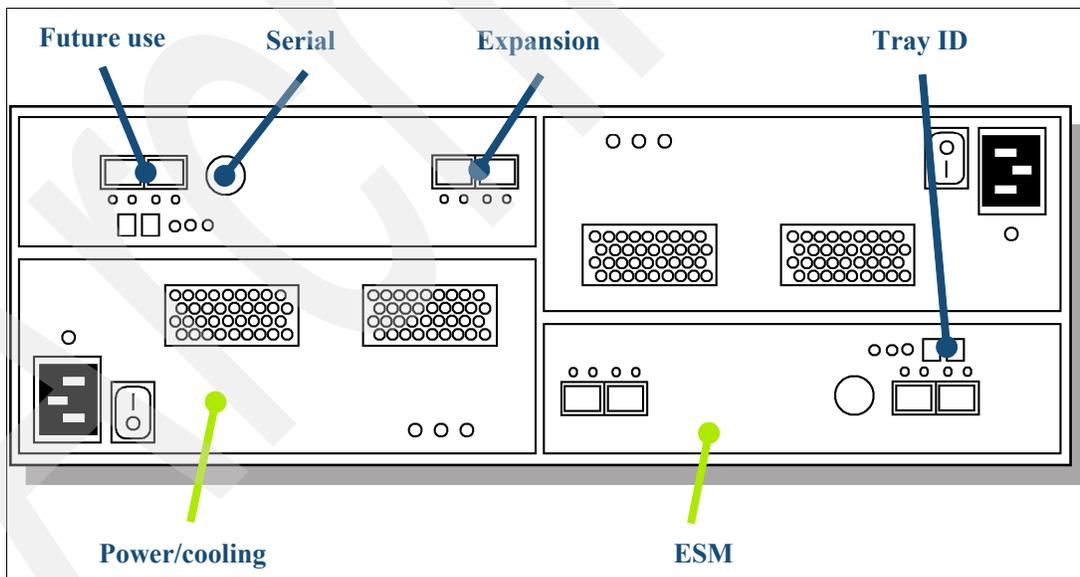


Figure 5-7 DS4000 EXP810 4 Gb Expansion Enclosure rear view

Cabling

The DS4700 Express storage system's four external drive loops are configured as two redundant pairs, with each pair cabling a maximum of three enclosures to the controller module. Figure 5-8 shows a fully configured DS4700 Express storage system with six DS4000 EXP810 Expansion Enclosures.

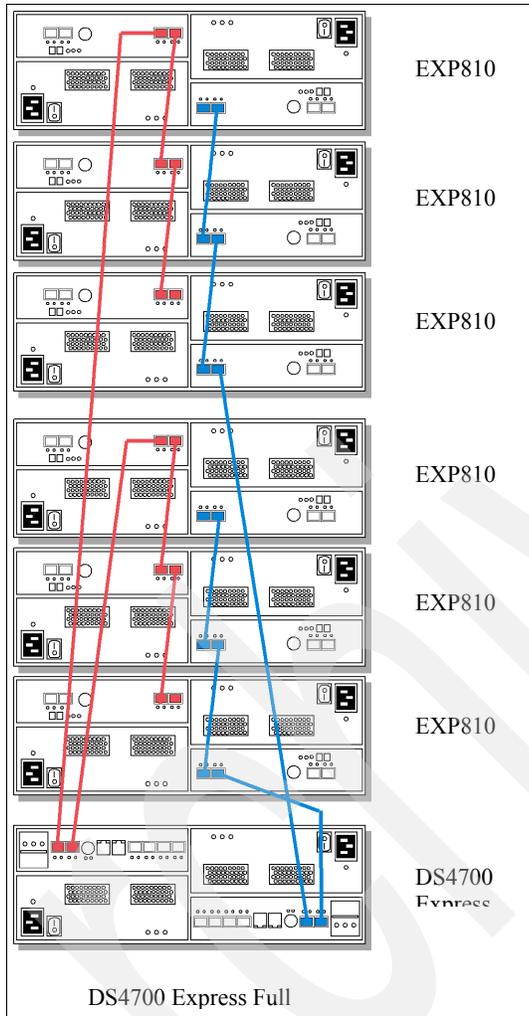


Figure 5-8 Fully configured DS4700 Express with six DS4000 EXP810 Expansion Enclosures

As noted earlier, the DS4700 Express storage system will support also 2 Gb disk drive modules.

For the highest availability, the DS4700 Express storage system utilizes a “top-down/bottom-up” cabling scheme to ensure access to available expansion units in the unlikely event that a full expansion unit is unavailable.

Most of the other modular storage systems on the market might use a simple daisy chain scheme, where both drive loops run from the controllers, to the first drive enclosure, then to the next, and so on. When cabling this way, one drive enclosure failure might result in access to all enclosures after it to be lost.

For more information about DS4700 Express, refer to the Web page:

<http://www.storage.ibm.com/disk/ds4000>

5.4 IBM N series (Network Attached Storage)

The IBM Storage System N Series (see Figure 5-9) provides a range of reliable, scalable storage solutions for a variety of storage requirements. These capabilities are achieved by using network access protocols such as NFS, CIFS, HTTP, and iSCSI as well as Storage Area technologies such as Fibre Channel. Utilizing built-in RAID technologies (either RAID-DP or RAID4, which will be fully described in a later chapter) all data is well protected with options to add additional protection through mirroring, replication, snapshots and backup. These storage systems are also characterized by simple management interfaces that make installation, administration, and troubleshooting uncomplicated and straightforward.

The IBM System Storage N Series is designed from the ground up as a standalone storage system.

5.4.1 Advantages of this storage solution

Advantages of using this type of flexible storage solution include the capability to:

- ▶ Tune the storage environment to a specific application while maintaining flexibility to increase, decrease, or change access methods with a minimum of disruption.
- ▶ React easily and quickly to changing storage requirements. If additional storage is required, being able to expand it quickly and non-disruptively is required. When existing storage exists but is deployed incorrectly, the capability to reallocate available storage from one application to another quickly and simply cannot be done.
- ▶ Maintain availability and productivity during upgrades. If outages are required, they can be kept to the shortest time possible.
- ▶ Create effortless backup/recovery solutions that operate commonly across all data access methods.
- ▶ File and block level services in a single system, helping to simplify your infrastructure.

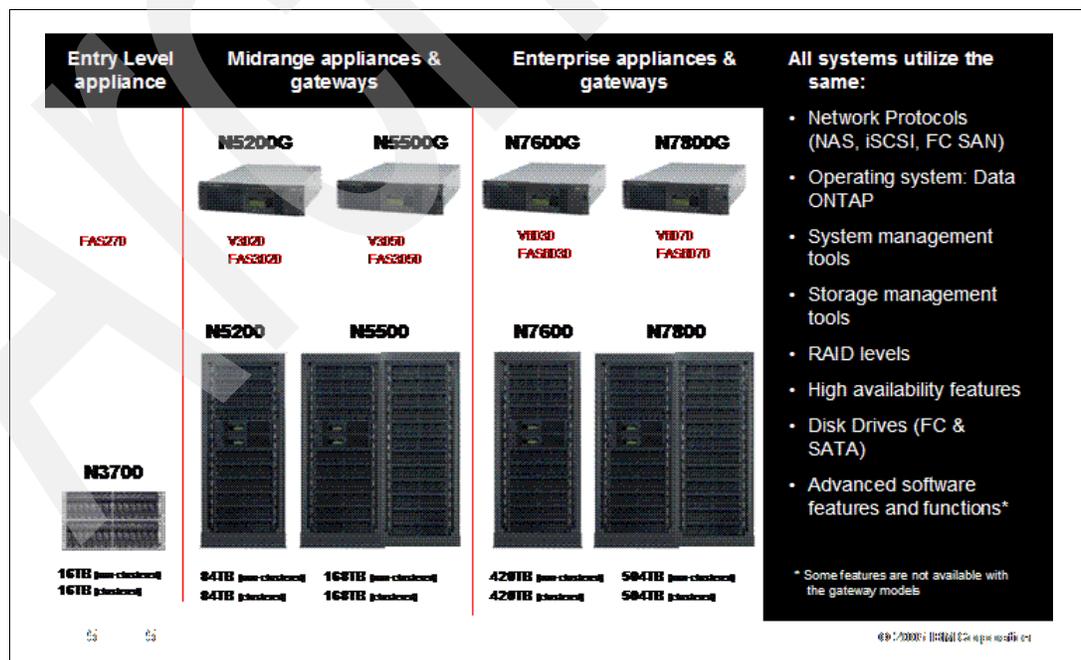


Figure 5-9 N series product line

5.4.2 The IBM N series standard software features

The standard software for the IBM ships or is enabled free of charge with the IBM N series product line (see Table 5-1).

Table 5-2 IBM N series standard software

Data ONTAP®	Data ONTAP is operating system software that optimizes data serving and allows multiple protocol data access.
FTP	File Transfer Protocol (FTP), a standard Internet protocol, is a simple way to exchange files between computers on the Internet.
Telnet	The TELNET Protocol provides a general, bi-directional, eight-bit byte oriented communications facility. It provides user oriented command line login sessions between hosts.
SnapShot	SnapShot enables online backups, providing near instantaneous access to previous versions of data without requiring complete, separate copies.
FlexVol	FlexVol creates multiple flexible volume on a large pool of disks. Dynamic, nondisruptive (thin) storage provisioning; space- and time-efficiency These flexible volumes can span multiple physical volumes without regard to size.
FlexCache	FlexCache has the ability to distribute files to remote locations without the necessity for continuous hands-on management. Filers deployed in remote offices automatically replicate, store, and serve the files or file portions that are requested by remote users without the necessity for any replication software or scripts.
Disk Sanitization	Disk sanitization is the process of physically obliterating data by overwriting disks with specified byte patterns or random data in a manner that prevents recovery of current data by any known recovery methods. This feature enables you to carry out disk sanitization by using three successive byte overwrite patterns per cycle. By default, six cycles are performed.
FilerView®	FilerView is a Web-based administration tool that allows IT administrators to fully manage N3700 systems from remote locations. It provides simple and intuitive Web-based single-appliance administration.
SnapMover®	SnapMover migrates data among N3700 clusters with no impact on data availability and no disruption to users.
AutoSupport	AutoSupport is a sophisticated, event-driven logging agent featured in the Data ONTAP operating software and inside each N series system which continuously monitors the health of your system and issues alerts if a problem is detected. These alerts can also be in the form of e-mail.
SecureAdmin™	SecureAdmin is a Data ONTAP module that enables authenticated, command-based administrative sessions between an administrative user and Data ONTAP over an intranet or the Internet.
DNS	The N series supports using a host naming file or a specified DNS server and domain.
Cluster Failover	<ul style="list-style-type: none"> ▶ Ensures high data availability for business-critical requirements by eliminating a single point of failure. ▶ Must be ordered for A20 clustered configurations or upgrades from A10 to A20 ▶ Active-active pairing delivers even more “nines to right of the decimal point”
NIS	The N series does provide NIS client support and can participate in NIS domain authentication.

Integrated automatic RAID manager	The IBM N series and Data ONTAP provide integrated RAID management with RAID-Double Parity (default) and RAID 4.
-----------------------------------	--

5.4.3 Optional software

The optional software for the IBM N series is fee based licensing on a individual basis (see Table 5-2).

Table 5-3 Optional software

CIFS	CIFS provides File System access for Microsoft Windows environments.
NFS	NFS provides File System access for UNIX and Linux environments.
HTTP	Hypertext Transfer Protocol allows a user to transfer displayable Web pages and related files.
FlexClone	FlexClone provides instant replication of data volumes/sets without requiring additional storage space at the time of creation.
Multistore	<ul style="list-style-type: none"> ▶ Permits an enterprise to consolidate a large number of Windows, Linux or UNIX file servers onto a single storage system. ▶ Many “virtual filers” on one physical appliance ease migration and multi-domain failover scenarios.
SnapLock	SnapLock provides non-erasable and non-rewritable data protection that helps enable compliance with government and industry records retention regulations.
LockVault	LockVault is designed to provide non-erasable and non-rewritable copies of Snapshot™ data to help meet regulatory compliance requirements for maintaining backup copies of unstructured data.
SnapMirror®	<ul style="list-style-type: none"> ▶ Remote mirroring software that provides automatic block-level incremental file system replication between sites. ▶ Available in synchronous, asynchronous and semi synchronous modes of operation.
SnapRestore®	SnapRestore allows rapid restoration of the file system to an earlier point in time, typically in only a few seconds.
SnapVault®	SnapVault provides disk based backup for N3700 systems by periodically backing up a snapshot copy to another system.
SnapDrive®	SnapDrive enables Windows and Unix applications to access storage resources on N series storage systems, which are presented to the Windows 2000 or later, operation system as locally attached disks. For Unix it allows you to create storage on a storage system in the form of LUNs, file systems, logical volumes, or disk groups.
SnapManager®	SnapManager provides host software for managing Exchange and SQL Server backup and restore. SnapManager software simplifies Exchange data protection by automating processes to provide hands-off, worry-free data management.
SnapValidator®	For Oracle deployments, SnapValidator can be used to provide an additional layer of integrity checking between the application and N series storage. SnapValidator allows Oracle to create checksums on data transmitted to N series storage for writes to disk and include the checksum as part of the transmission.

5.4.4 IBM System Storage N3700 Introduction

The N3700 Filer is a 3U solution designed to provide NAS and iSCSI functionality for entry to mid-range environments. The basic N3700 offering is a single-node model A10, which is upgradeable to the dual-node model A20 and requires no additional rack space. The dual-node, clustered A20, is designed to support fail over and fail back functions to maximize reliability. The N3700 filer can support 14 internal hot-plug disk drives with scalability being provided through attachment to up to three 3U EXN2000 expansion units, each with a maximum of 14 drives. The N3700 also has the capability to connect to a Fibre Channel tape for backup.

5.4.5 N5200 and N5500 Models A10 and A20

The N5200 and N5500 are suitable for environments that demand data in high availability, high capacity and highly secure data storage solutions. The IBM System Storage N5000 series offers additional choice to organizations for enterprise data management. The IBM System Storage N5000 series is designed to deliver high-end enterprise storage and data management value with midrange affordability. Built-in enterprise serviceability and manageability features help support your efforts to increase reliability, simplify and unify storage infrastructure and maintenance, and deliver exceptional economy.

The IBM N5000 A series comes in two models:

- ▶ N5200
 - 2864-A10 Single Filer
 - 2864-A20 Clustered
- ▶ N5500
 - 2865-A10 Single Filer
 - 2865-A20 Clustered

5.4.6 N5000 series gateway

The IBM System Storage N 5000 series Gateway, an evolution of the N 5000 series product line, is a network-based virtualization solution that virtualizes tiered, heterogeneous storage arrays, allowing customers to leverage the dynamic virtualization capabilities available in Data ONTAP across multiple tiers of IBM and 3rd party storage. Like all N series storage systems, the N series Gateway family is based on the industry-hardened Data ONTAP microkernel operating system, which unifies block and file storage networking paradigms under a common architecture and brings a complete suite of N series advanced data management capabilities for consolidating, protecting, and recovering mission-critical data for enterprise applications and users.

The N series Gateway offers customers new levels of performance, scalability and a robust portfolio of proven data management software for sharing, consolidating, protecting, and recovering mission critical data. N series storage systems seamlessly integrate into mission-critical SAN environments and provide a simple, elegant data management solution decreasing management complexity, improving asset utilization, and streamlining operations to increase business agility and reduce total cost of ownership.

Organizations that are looking for ways to leverage SAN-attached storage to create a consolidated storage environment for the various classes of applications and storage requirements throughout their enterprise. These prospects are looking for ways to increase utilization, simplify management, improve consolidation, enhance data protection, enable rapid recovery, increase business agility, deploy heterogeneous storage services and broaden centralized storage usage by provisioning SAN capacity for business solutions requiring NAS, SAN or IP SAN data access.

These prospects have:

- ▶ Significant investments or a desire to invest in a SAN architecture
- ▶ Excess capacity and/or an attractive storage cost for SAN capacity expansion
- ▶ Increasing requirements for both block (FCP, iSCSI) and file (NFS, CIFS) access
- ▶ Increasing local and/or remote shared file services and file access workloads.

They are seeking solutions to cost effectively increase utilization; consolidate distributed storage, Direct Access Storage and file services to SAN storage; simplify storage management; and improve storage management business practices

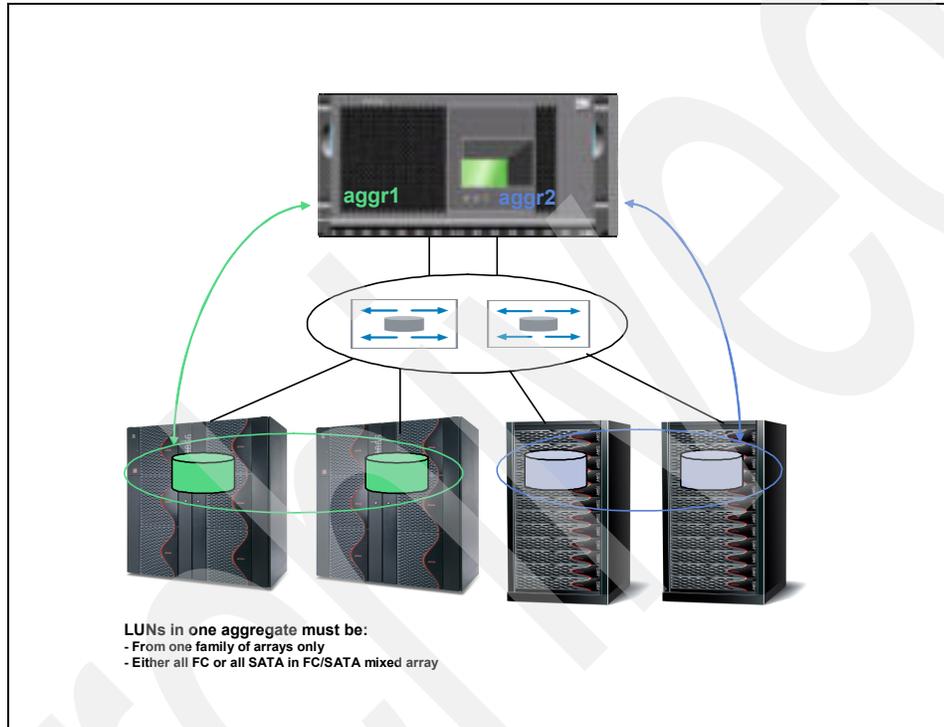


Figure 5-10 Heterogeneous storage

IBM N series Gateway highlights

IBM System Storage N series Gateway provides a number of key features that enhance the value and reduce the management costs of utilizing a Storage Area Network (SAN).

An N series Gateway has the following capabilities:

- ▶ Simplifies storage provisioning and management
- ▶ Lowers storage management and operating costs
- ▶ Increases storage utilization
- ▶ Provides comprehensive simple-to-use data protection solutions
- ▶ Improves business practices and operational efficiency
- ▶ Transforms conventional storage systems into a better managed storage pool (see Figure 5-11).

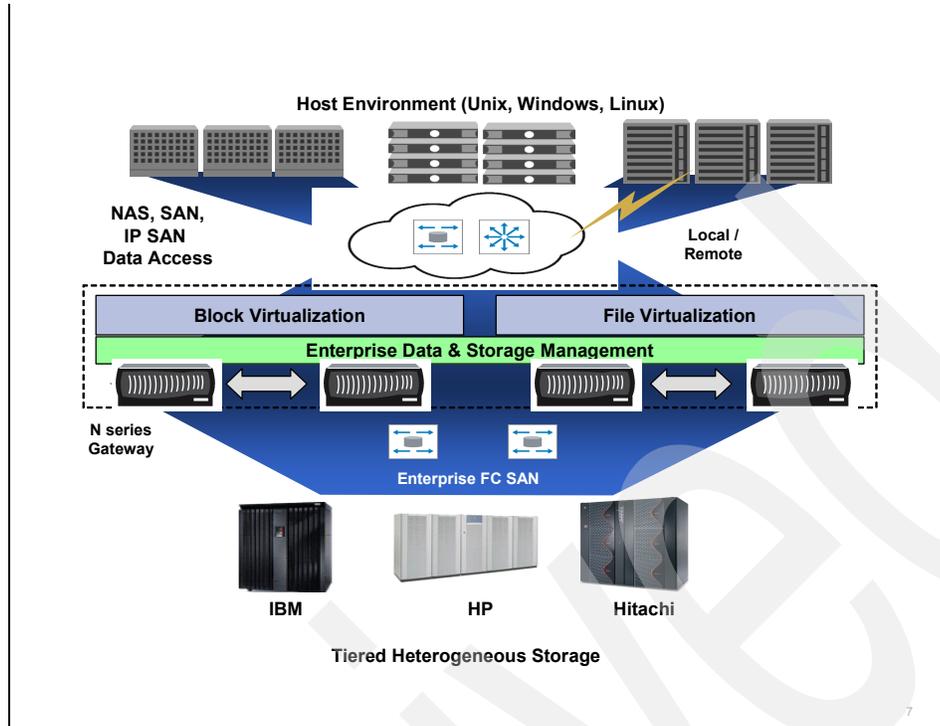


Figure 5-11 Tiered Heterogeneous Storage

What is an N series Gateway?

The N series Gateway is a network-based virtualization solution that virtualizes tiered, heterogeneous storage arrays and enables customers to leverage the dynamic virtualization capabilities of Data ONTAP software across a broad set of high-end and modular storage arrays from Hitachi, HP, IBM, Engenio, StorageTek™, and Sun.

Industry's most comprehensive virtualization solution, the N series Gateway s provides proven and innovative data management capabilities for sharing, consolidating, protecting, and recovering mission-critical data for enterprise applications and users and seamlessly integrates into mission-critical enterprise-class SAN infrastructures. These innovative data management capabilities when deployed with disparate storage systems simplify heterogeneous storage management.

The N series Gateway will present shares, exports or LUNs that are built on flexible volumes which reside on aggregates. The N series Gateway is also a host on the storage array SAN. Disks are not shipped with the N series Gateway. N series Gateways take storage array LUNs (which are treated as disks) and virtualize them through Data ONTAP, presenting a unified management interface.

Gateway models

The following models are available:

- ▶ N5500:
 - 2865-G20 (cluster)
- ▶ N5200:
 - 2864-G10
 - 2864-G20 Clustered model

- ▶ N5500:
 - 2865-G10
 - 2865-G20 Clustered model

5.5 Optical storage

The IBM 3996 optical library is an externally attached, optics storage library that uses 30 GB optical disc technology. The 3996 library is offered in three models and is available for attachment to most models of the IBM i5 and iSeries family of workstations and servers. Figure 5-12 shows the three models of the IBM 3996.



Figure 5-12 The three models of IBM 3996 optical library

This family of optical libraries features 5.25 inch, 30 GB Ultra™ Density Optical (UDO) technology, and the UDO media provides up to five times the maximum capacity of media used in the previous 3995 optical library offered by IBM. The IBM 3996 Optical Library supports permanent Write Once / Read Many (WORM), and rewriteable recording technologies in a single library. The IBM 3996 is available as a low voltage differential (LVD) SCSI interface connectivity and has an optional barcode scanner to facilitate library inventory.

The 3996 Optical Library is offered in three models; Model 032, Model 080, and the Model 174. Each model supports permanent Write Once / Read Many (WORM), and rewriteable recording technologies in a single library:

- ▶ The Model 32 has the ability to handle up to 32 disks, providing up to 960 GB of physical capacity. The Model 32 has one optical disc drive, and an option for adding a second drive.
- ▶ The Model 80 has the ability to handle up to 80 disks, providing up to 2.4TB of physical capacity. The Model 80 has two optical disc drives with an option of increasing to four drives. When additional drives are added, the Model 80 has the ability to handle up to 72 disks, providing up to 2.16TB of physical capacity.
- ▶ The Model 174 has a physical capacity of up to 5.2TB; each of the one hundred and seventy four media slots holds a disk with up to 30 GB of optical storage. The Model 174 has two optical disc drives with an option of increasing to four drives. When the additional drives are added, the 3996 Model 174 has the ability to handle up to 166 disks, providing up to 4.98TB of physical capacity.

The IBM 3996 features an optional barcode scanner in all three optical model offerings. The three main benefits of bar coding are out-of-library media management, faster media inventoring inside the library, and added security.

5.6 Tape storage

While sometimes tape is referred to as obsolete, new retention requirements have made tape interesting again. Disaster recovery solutions more and more tend to prefer disk, virtual tape, or disk to tape solutions.

IBM offers two tape classes, Linear Tape-Open (LTO) Ultrium and IBM 3592. For each class, a Read/Write cartridge and a Write Once Read Many (WORM) cartridge is available.

5.6.1 LTO Ultrium tape drive

The Linear Tape-Open (LTO) program was conceived as a joint initiative of IBM, Hewlett-Packard, and Seagate Technology. In 1997, the three technology provider companies set out to enable the development of best-of-breed tape storage products by consolidating state-of-the-art technologies from numerous sources, and in November of that year they produced a joint press release about LTO. The three technology provider companies for LTO are HP, IBM Corporation, and Certance LLC (now owned by Quantum).

The three LTO sponsoring companies also took steps to protect customer investment by providing a four-generation roadmap, shown in Figure 5-13, and establishing an infrastructure to enable compatibility between products. At the time of writing, LTO generations 1, 2, and 3 are available.

	Generation 1	Generation 2	Generation 3	Generation 4	Generation 5	Generation 6
Capacity (Native)	100 GB	200 GB	400 GB	800 GB	1.6 TB	3.2 TB
Transfer Rate (Native)	Up to 20 MB/s	Up to 40 MB/s	Up to 80 MB/s	Up to 120 MB/s	Up to 180 MB/s	Up to 270 MB/s
WORM	No	No	Yes	Yes	Yes	Yes

Figure 5-13 LTO Ultrium roadmap

Important: Hewlett-Packard, IBM, and Certance reserve the right to change the information in this migration path without notice.

The LTO Ultrium compatibility investment protection is provided based on these principles:

- ▶ An Ultrium drive is expected to read data from a cartridge in its own generation and at least the two prior generations.
- ▶ An Ultrium drive is expected to write data to a cartridge in its own generation and to a cartridge from the immediately prior generation in the prior generation format.

Next we discuss compatibility between available Ultrium 1, Ultrium 2, and Ultrium 3 media.

IBM Ultrium 1, 2, and 3 compatibility

IBM Ultrium 2 tape drives (both standalone and in IBM Ultrium libraries) support both Ultrium 1 and Ultrium 2 cartridges. An Ultrium 1 cartridge in an Ultrium 2 drive will be written at the same 100 GB native capacity, but with improved performance (20 MB/s). Ultrium 1 drives cannot read or write an Ultrium 2 cartridge. If you put an Ultrium 2 cartridge in an Ultrium 1 drive, then you will get an “Unsupported Cartridge Format” failure.

Similarly, the Ultrium 3 drive reads and writes Ultrium 2 cartridges, and also reads Ultrium 1 cartridges. The Ultrium 3 cartridge can only be used by the Ultrium 3 drive. This is in accordance with the LTO design specifications. Figure 5-14 shows the compatibility.

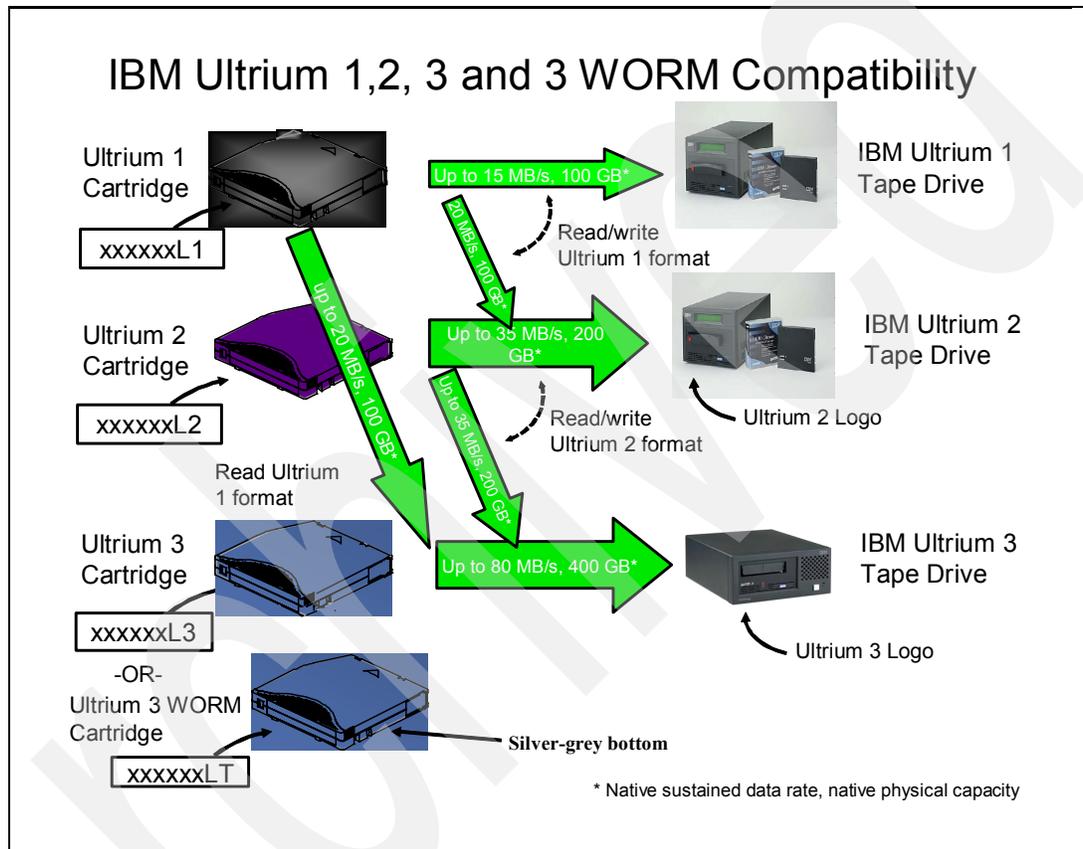


Figure 5-14 IBM Ultrium 1, 2, and 3 compatibility

IBM System Storage TS1030 Tape Drive

The new TS1030 LTO Tape Drive is designed for installation in:

- ▶ IBM System Storage TS3500 Tape Library models L53 and D53
- ▶ IBM TotalStorage 3584 Tape Library models L52, L32, D52, and D32

Note: The LTO Ultrium 3, 4 Gb Fibre Channel drive is also available as a feature code in the TS3100 Tape Library, TS3200 Tape Library, and TS3310 Tape Library.

The TS1030 LTO Tape Drive drive offers high capacity, performance, and technology designed for the midrange open systems environment. The TS1030 LTO Tape Drive has a 4 GB Fibre Channel interface for either point-to-point or Fibre Channel-Arbitrated Loop attachment.

The native data transfer is 80 MB/sec and it uses the IBM TotalStorage LTO Ultrium 400 GB data cartridge or up to 800 Gb with a 2:1 compression.

The TS1030 LTO Tape Drive uses the new dual-stage 16 head actuator for a more precision head alignment to help support higher track density and improved data integrity and a new independent tape loader and threader motors with positive pin retention. The new pin retention mechanism prevent stretching of breaking the tape and loose tape wraps. Also the tape loader and threader motors are designed to help improve the reliability of loading and unloading a cartridge, and to retain the pin even if the tension is dropped. The TS1030 LTO Tape Drive has a 128 MB internal buffer.

Some highlights of the TS1030 LTO Tape Drive are discussed in the following sections.

Dynamic breaking

The TS1030 LTO Tape Drive uses dynamic breaking. In the event of power failure, reel motors are designed to maintain tension and gradually decelerate instead of stopping abruptly, reducing the tape breakage, stretching, or loose tape wraps during a sudden power-down.

Servo and track layout technology

The TS1030 LTO Tape Drive uses 704 data tracks the read and write to tape. High bandwidth servo system features a low-mass servo to help more effectively track servo bands and improve data throughput with damaged media in less-than-optimal shock and vibration environments.

Surface Control Guiding Mechanism

The Surface Control Guiding Mechanism is designed to guide the tape along the tape path in the S1030 LTO Tape Drive. This method uses the surface of the tape, rather than the edges, to control tape motion. This helps to reduce tape damage (especially to the edges of the tape) and tape debris, which comes from the damaged edges and can accumulate in the head area.

Magneto Resistive (MR) head design

This design is using a flat lap head technology in MR heads for Ultrium 3 that helps to minimize the contact, debris accumulation, and wear on the tape as it moves over the read/write heads.

Dynamic Amplitude Asymmetry Compensation

This design helps to dynamically optimize readback signals for linear readback response from magneto resistive read head transducers.

5.6.2 3592 J1A and TS1120 tape drives

The IBM TotalStorage 3592 Tape Drive Model J1A and the System Storage TS1120 Tape Drive offer a solution to address applications that require high capacity, fast access to data or long-term data retention. It is supported in IBM tape libraries or frames that support stand-alone installations, and is supported in an IBM 3592 C20 frame attached to a StorageTek 9310 library. It is designed to help reduce the complexity and cost of the tape infrastructure.

Technology

The TS1120 tape drive provides up to 60% more capacity and 150% more performance than the IBM TotalStorage 3592 J1A tape drive that it supersedes, and more than eight times the capacity and seven times the performance of the IBM TotalStorage 3590 H1A tape drive. The tape drive uses the existing 3592 media, which is available in re-writable or Write Once Read Many (WORM) media to store 100 GB or 500 GB depending on cartridge type. The 3592 JA/JW media helps reduce resources to lower total costs, while the 3592 JJ/JR media is designed to support applications that require rapid access to data.

In an open systems or mainframe environment, the TS1120 tape drive can use the 3592 JJ cartridge or format a 3592 JA cartridge to a 100 GB capacity to reduce the average locate time using a unique function called Capacity Scaling. Tape drives can be shared among supported open system hosts on a Storage Area Network (SAN), or between FICON and ESCON hosts when attached to a 3592 J70 controller. This optimizes drive utilization and helps reduce infrastructure requirements.

High performance

The TS1120 tape drive supports a native data transfer rate of up to 100 MB/s. In open system environments where data typically compresses at 2:1, the TS1120 tape drive can transfer data up to 200 MB/s. In a mainframe environment where data typically compresses at 3:1, a single tape drive can transfer data up to 144 MB/s. This can help reduce backup and recovery times or require fewer resources to support the environment.

3592 cartridge and media

The 3592 and TS1120 tape drive support four types of the IBM TotalStorage Enterprise Tape Cartridge 3592, two rewriteable (R/W) types (JA & JJ) and two Write Once Read Many (WORM) types (JW & JR). There are two of each kind (R/W and WORM) to make available both a full length version and a Short Length Cartridge (SLC™) version. Specifically the JA and JW are the full length and capacity types, and the JJ and JR are the SLC types, of the R/W and WORM cartridges, respectively.

All four types have the same physical outline, or form factor, which is similar to that of the 3590 tape cartridge, and which consequently allows them to be used in the IBM TotalStorage Enterprise Tape Library 3494 and StorageTek Automated Cartridge System (ACS) solutions that can handle the 3590 tape cartridge. Additionally, the IBM TotalStorage Tape Library 3584 supports 3592 cartridge types. The four types of 3592 cartridge all contain tape media with a new dual-coat, advanced-particle rewriteable magnetic media (the WORM characteristic of the JW and JR cartridge types is achieved by other means, as we discuss in the following sections).

This is a new type of media that has improved areal density capabilities and differs from the tape media in any previously shipped IBM branded cartridge. The media is housed in a cartridge shell, which is close, but not identical, to current 3590 cartridges in size and shape. The new 3592 cartridge was designed to have the strength and durability of an enterprise cartridge. Enhanced assembly strengthens the cartridge at critical locations and helps make the 3592 cartridge less susceptible to damage (for example, if dropped) than would otherwise be the case. These features help create an extremely strong and durable cartridge, both within an automated tape library and when (mis)handled by humans.

The four cartridge types each have a unique label which clearly identifies which type it is. Beyond that, the WORM cartridge types are readily distinguishable at a distance from the R/W cartridge types by means of the color of the cartridge shell. The WORM cartridge types have a platinum colored cartridge shell, the R/W cartridge types have a black shell.

Additionally, the SLC cartridge types (JJ and JR), beyond having tape which is physically shorter than the full length tapes, are readily distinguished from the full length cartridges at a distance by the color of the cartridge accouterments: the sliding door and the locking mechanism. The accouterments of the SLC cartridge types are light blue, those of the full length cartridge types are darker blue. Aside from the differences in labels, color of the cartridge shells, accouterments, and in physical length of the tape enclosed, the cartridges are otherwise identical and are described generically as follows when their differences are not relevant.

The tape is pulled from the cartridge by means of a leader pin rather than a leader block as in the 3590. A sliding door covers the area formerly occupied by the leader block in a 3590 cartridge, and is pushed back by the loader mechanism when the cartridge is loaded, so that the leader pin can be accessed, and the tape within the cartridge drawn out. A locking mechanism prevents the media from unwinding when the cartridge is not located within a drive. There are other cartridge features which prevent it from being inserted into a 3590 or inserted into a 3592 in an improper orientation.

Contained within the cartridge is the Cartridge Memory (CM), which is a passive, contactless silicon storage device that is physically enclosed by the cartridge shell. The CM is used to hold information about that specific cartridge, its type, the media in the cartridge, and the data on the media. The 3592 Tape Drive uses the same CM as LTO Ultrium media, with a capacity of 4,096 bytes. However, it is important to note that the format of the CM has been redesigned for the 3592 to support certain advanced features which are not included in the LTO specification.

Cartridge capacity

IBM TotalStorage Enterprise Tape Cartridges 3592 are designed to work with the first-generation IBM TotalStorage Enterprise Tape Drive 3592 Model J1A (3592 J1A tape drive) and the second-generation IBM System Storage TS1120 Tape Drive (TS1120 Tape Drive). Cartridges are available in two lengths and in either re-writeable or Write Once, Read Many (WORM) formats. The short length 3592 JJ/JR cartridges provide rapid access to data and the standard length 3592 JA/JW cartridges provide high capacity.

Rewriteable cartridges

The first-generation 3592 J1A tape drive can initialize short length JJ cartridges to 60 GB and initialize (or re-initialize) standard JA length cartridges to either 60 GB (to support fast time to data) or 300 GB (to support high capacity).

The second-generation TS1120 tape drive can initialize short length JJ cartridges to 60 or 100 GB and initialize (or re-initialize) standard length JA cartridges to 60, 100, 300, or 500 GB to support fast access to data or to help address data growth and facilitate interchange. At typical compression ratios, the 3592 JA cartridge can provide usable capacity of up to 1TB in an open system environment, and up to 1.5 TB in an IBM System z9™ environment when used with a TS1120 Tape Drive. The JA and JJ cartridge models are suitable for storing data that has a finite life span and are rewriteable.

WORM cartridges

The TS1120 and 3592 J1A tape drives are designed to work with Write Once, Read Many (WORM) JR and JW cartridges to store data in a non-erasable, non-rewriteable format. This is intended to help support the long term retention of reference data and meet the requirements of regulatory bodies worldwide. The short length JR and standard length JW cartridges have advanced security features that are designed to prevent the alteration or deletion of stored data while allowing data to be appended to existing cartridges or files.

The WORM cartridge types are geometrically identical to the R/W cartridge, and uses the same rewriteable media formulation. The servo format which is mastered onto the tape at manufacturing is different for WORM cartridge types however. The WORM aspect comes not from any inherent non-reversible media characteristic, but rather by the way the WORM firmware will handle a WORM cartridge.

The WORM firmware is designed to prevent over-write or erase of previously written customer data such as Records or File Marks, though some Records and File Marks which are readily identifiable as constructs put by applications around customer data (for example, trailer labels) might be overwritten if no customer data follows, which allows use of existing applications (for example, which append data to existing files).

In that the media is inherently rewriteable, WORM functionality is achieved through drive controls, just as is done in WORM tape offerings offered by other vendors. The intent is to be a transparent replacement for other enterprise WORM tape offerings from an application software point of view.

The drive firmware determines whether the cartridge is R/W or WORM and then operates accordingly. This determination is continuously validated to make it very difficult for anyone attempting to tamper with. If the determination is that the cartridge is WORM, then WORM functionality is exhibited. The design is to only exhibit standard R/W functionality if the determination is that the cartridge is unequivocally R/W. If there is any evidence of tampering, the drive appropriately controls access to (for example, write fences) the media.

Data compression

3590 customers have become accustomed to Adaptive Lossless Data Compression (ALDC). The 3592 Tape Drive uses the same Streaming Lossless Data Compression Algorithm (SLDC) used in IBM LTO products which achieves the same, or in some cases (for example, incompressible data) better, data compression than does ALDC. In particular, SLDC does not expand incompressible data as did ALDC, therefore, there is no requirement to disable data compression when recording scientific, image, precompressed, or encrypted data which is not compressible.

The 300/500 GB native capacity of the full length cartridge types is achieved by recording data in a linear serpentine pattern over a user area 570 meters in length, the approximate length of a 3590 Extended Length cartridge. This configuration is advantageous in many, but not all, customer environments. There are environments where different types of performance enhancements are valued much more than is capacity. To suit the requirements of customers with these types of environments, several options are supported including capacity scaling, segmentation, and SLC cartridge types.

Capacity scaling and segmentation

Capacity scaling, which is only enabled on the JA cartridge type, allows a customer to logically reduce the cartridge capacity of a tape if he is willing to trade capacity away for performance. A customer can capacity scale a JA cartridge by sending a Mode Select command to it with essentially a one byte argument. Alternately a customer can buy a JA full length cartridge capable of 300/500 GB already capacity scaled down to one of two other capacity points: 60/100 GB or 260 GB. Some capacity scaling settings cause the drive to change the way data is stored to tape in interesting ways other than simply shortening the length of tape recorded on.

5.6.3 Tape automation

Tape drives and cartridges can be used standalone and in tape automation solutions.

IBM System Storage TS3100 Tape Library

The TS3100 Tape Library, (machine type 3573), provides a single Ultrium 3 tape drive and holds a total of 22 cartridges (8.8 TB native capacity) in two removable magazines.

This entry level desktop or a rack mounted unit (requiring two rack units of a industry standard 19 inch rack) can be operated in random or sequential mode, permitting unattended backup operations. A single dedicated mail slot (I/O Station) is available for importing and exporting cartridges.

The Ultrium 3 tape drive is available with one of the following two interfaces:

- ▶ SCSI LVD
- ▶ 4 GB Native Fibre Channel

The Ultrium 3 media has a native capacity of 400 GB and the Ultrium 3 tape drive provides a sustained maximum transfer rate of 80 MB/s.

Standard features are a bar code reader and a remote management through a Web User Interface.

Figure 5-15 shows the front view of the TS3100 Tape Library. The I/O station is located in the lower left storage magazine. In the middle of the TS3100 Tape Library is the Operator Control Panel.



Figure 5-15 Front view of the TS3100 Tape Library

IBM System Storage TS3200 Tape Library

The TS3200 Tape Library, (machine type 3573), provides two Ultrium 3 tape drives and holds a total of 44 cartridges (17.6 TB native capacity) in four removable magazines.

This entry level desktop or a rack mounted unit (requiring four rack units of a industry standard 19 inch rack) can be operated in random or sequential mode, permitting unattended backup operations. Three mail slots (an I/O Station) are available for importing and exporting cartridges.

Two Ultrium 3 tape drives can be installed. Each one can have either of the following two interfaces:

- ▶ SCSI LVD
- ▶ 4 GB Native Fibre Channel

Note: This library can be partitioned into two logical libraries, each with one tape drive and all of the storage slots in the magazines on each side. If two tape drives are installed with different interfaces, the library *must* be partitioned.

The Ultrium 3 media has a native capacity of 400 GB, and the Ultrium 3 tape drive provides a sustained maximum transfer rate of 80 MB/s.

Standard features are a bar code reader and a remote management through a Web User Interface.

Figure 5-16 shows the front view of the TS3100 Tape Library. The three slot I/O station is located in the lower left storage magazine. In the middle of the TS3100 Tape Library is the Operator Control Panel.

Optionally, this library can also provide:

- ▶ A second power supply for redundancy
- ▶ Control path and Data path failover



Figure 5-16 Front view of the TS3200 Tape Library

IBM System Storage TS3310 Tape Library

The TS3310 Tape Library is a highly expandable Ultrium LTO3 library which allows you to start small with a 5U base module available in desktop or rack mounted configurations. Over time, as your requirement for tape backup expands, you can add additional 9U expansion modules, each of which contains space for additional cartridges, tape drives and a redundant power supply. The entire system grows vertically. Currently, available configurations include the 5U base library module alone or with up to two 9U modules. Future configurations will allow the 5U base module to be expanded with an additional four 9U modules.

The TS3310 Tape Library offers a broad range of configuration possibilities. The smallest configuration includes a base unit with one to two LTO3 tape drives, 12 TB of native tape storage (30 slots) and 6 I/O slots. This will be upgradeable to a fully configured rack mounted library 41U high with up to 18 LTO3 tape drives, over 158 TB of native tape storage (396 slots) and up to 48 I/O slots.

Figure 5-17 shows how the base module can be expanded.

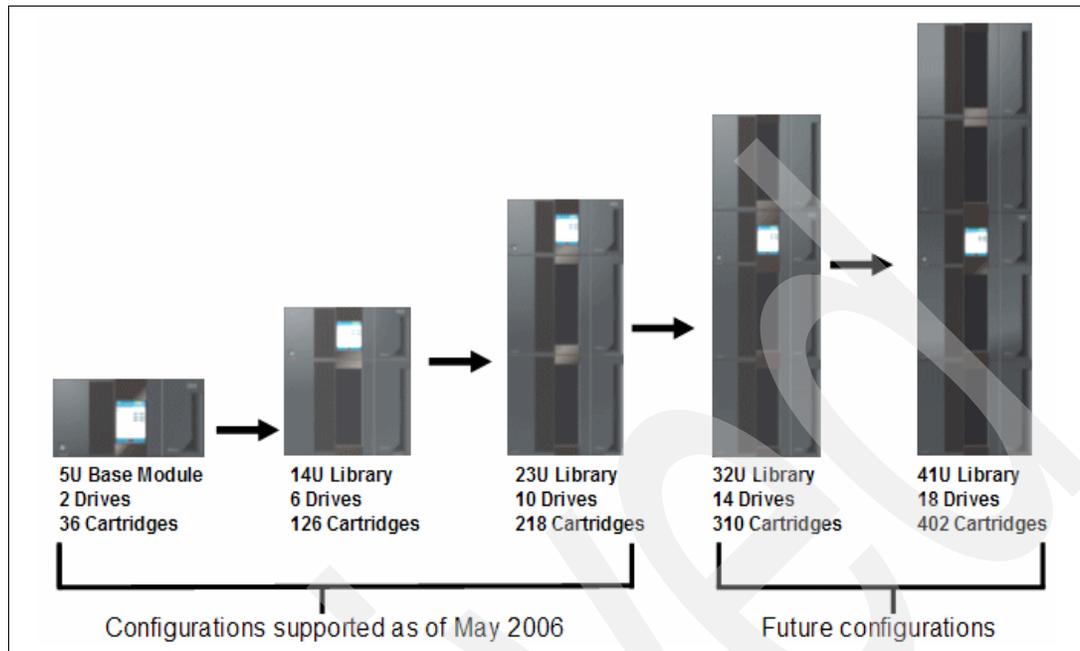


Figure 5-17 TS3310 configuration options

As with all IBM Ultrium Tape Libraries with more than one tape drive, the library can be partitioned into logical libraries. A bar code reader and remote management via a Web User Interface is standard.

Optionally, this library can also provide:

- ▶ A second power supply for redundancy
- ▶ Control path and Data path failover
- ▶ Up to two expansion modules (four will be supported in the future)
- ▶ Two power supplies in each module for redundancy

IBM System Storage TS3500 Tape Library

The IBM TS3500 Tape Library (machine type 3584) is a modular tape library consisting of frames that house tape drives (both LTO3 and 3592) and cartridge storage slots.

This IBM Tape Library library offers the greatest:

- ▶ Expendability
- ▶ Availability
- ▶ Automated operations

Expendability

You can install a single-frame base library (see Figure 5-18) and grow it up to 16 frames (see Figure 5-19) tailoring the library to match your system capacity and performance requirements from 13 TB to 2755 TB (up to 8265 TB with 3:1 compression), and using from one to 192 IBM tape drives.

Figure 5-18 shows a single frame TS3500 Tape Library from the front. The Operator Control Panel and I/O door can be seen in the front door.



Figure 5-18 Single frame TS3500 Tape Library

Figure 5-19 shows a fully expanded TS3500 Tape Library from the front left. Note that all expansion frames are added to the right of the base frame.

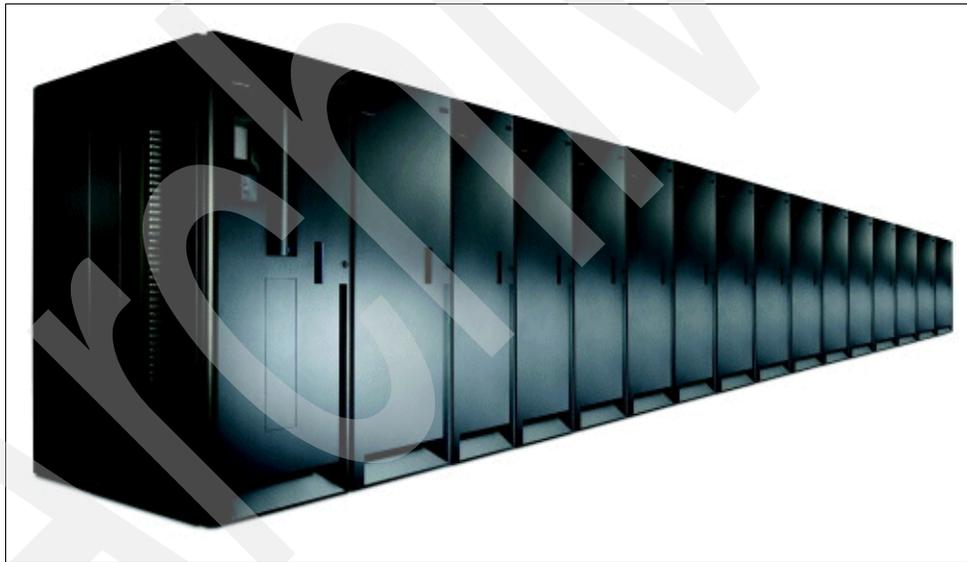


Figure 5-19 Fully expanded 16 frame TS3500 Tape Library

Availability

In addition to providing the most expansion potential, the IBM TS3500 Tape Library also provides the greatest degree of redundancy and availability features to maximize uptime, including these features:

- ▶ Redundant power supplies in every frame
- ▶ An optional second cartridge accessor
- ▶ Each cartridge accessor having dual grippers
- ▶ Redundant library control and data paths and load balancing over data paths
- ▶ Non-disruptive library and drive firmware updates
- ▶ Persistent world wide names for hot swappable tape drives that are replaced

The TS3500 Tape Library also provides proactive maintenance functionality at no extra charge. Through a service referred to as “Call Home”, the library will automatically contact the IBM Support Center if a problem occurs. It will open a Problem Management Record with details of errors and provide logs to aid in problem diagnosis and identification of spare parts that might be required.

Automated operations

With Advanced Library Management System and “Virtual I/O”, media import and export is greatly simplified.

- ▶ Import: Cartridges placed in the bulk I/O are now automatically moved into the body of the library. This allows an operator to load as many as 20 cartridges into a library with a 16 port I/O station without having to contact the Storage Management Administrator to request that the media be moved in.

Note: The library will move the cartridges into the body of the library as a convenience. From the Storage Application’s perspective, the media is still in an I/O slot. It is therefore necessary at some point to run the same commands as usual to make those tapes available (in Tivoli Storage Manager this would be `checkin libvolume`). When the command is run, the tapes are not moved, but the nature of the slot changes from being a virtual I/O slot to being a normal storage slot.

- ▶ Cartridge assignment into logical libraries: If the library is partitioned, cartridges are automatically assigned to different logical libraries according to predefined rules regarding the volume labels. In simpler libraries the media operator would have to specify which cartridges are to be assigned to which partition. Now the operator only has to load the media.
- ▶ Export: Cartridge export is also much simpler as the application can now request that more tapes are moved out of the library than there are I/O slots available. Storage application administrators can request that 20 cartridges are ejected even if there are only 16 I/O slots available. ALMS handles the effective queuing of media ejections transparently to the application. After the media operator has removed the first 16 cartridges, the remaining media is ejected.

5.7 Virtualization solutions

What is storage virtualization? There is not a single answer to this question, but many answers, depending on what storage aspect you are analyzing.

For example, a file system can be considered a virtualization layer because it creates an abstraction, or virtualization layer, between the application requesting a file by name and the file’s location on a physical storage that is ultimately described by some set of coordinates such as logical unit number, the LUN or disk, and relative block address to the data and length. Therefore, the LUN or disk is the physical object.

Now consider that the LUN resides on an enterprise class storage subsystem. With all probability, the LUN will be a virtual object that is mapped to one or more physical disk devices, using some kind of RAID protection, by the storage subsystem logic.

Therefore, now we have two levels of virtualization between the file and storage, and there are other levels as well, which we will not discuss here.

We can also consider another virtualization example: when disk is tape, and when tape is disk. This might look like wordplay but it refers to two very real virtualization products.

Hierarchical storage management (HSM) solutions offer transparent file movement between different storage tiers. When an old file migrates to tape the application continues to see the file on the disk file system, and when the application accesses the data, the file can be recalled or in some cases accessed directly on a tape storage device. Therefore, the application thinks it is accessing disk when it really is accessing a tape device.

The opposite example is relative to tape virtualization products such as the IBM TS7510 Virtualization Engine™ for tape. The application will see a tape drive, mount a volume and write to it and then dismount it. From the application perspective, all data has been written to tape but the TS7510 emulates tape and writes the data to disk.

Therefore, why do we do this? Why virtualize storage, and what does it have to do with ILM?

Virtualization can assist us in managing the underlying storage more efficiently, drive up storage utilization, and simplify data movement between storage tiers. Or we might have a legacy application that only supports offline storage such as tape and we want to put the resulting data on disk.

We will introduce the following storage virtualization products and position them as ILM tools:

- ▶ IBM TotalStorage SAN Volume Controller
- ▶ IBM Virtualization Engine TS7510
- ▶ IBM TotalStorage 3494 Virtual Tape Server (VTS)

5.7.1 IBM TotalStorage SAN Volume Controller

The SAN Volume Controller (SVC) (see Figure 5-20) is designed to simplify your storage infrastructure by enabling changes to the physical storage with minimal or no disruption to applications.



Figure 5-20 SVC

SAN Volume Controller combines the capacity from multiple disk storage systems into a single storage pool, which can be managed from a central point. This is simpler to manage, and it helps to increase utilization and improve application availability.

SAN Volume Controller's extensive support for non-IBM storage systems, including EMC, HP, and HDS, enables a tiered storage environment to better allow you to match the cost of the storage to the value of your data. It also allows you to apply advanced copy services across storage systems from many different vendors to help further simplify operations.

From a technical point of view, the SVC externalizes LUNS to servers. These LUNs are called *VDisks*, where V stands for *virtual*. Application servers or hosts access LUNS, and these LUNs are none other than VDisks.

The VDisks are mapped to *MDisks*, where M stands for *managed*, and the mapping is flexible, for example, an MDisk can be mapped to multiple VDisks and multiple MDisks can be combined into one VDisk. VDisks can be transparently migrated between different MDisks without disrupting application access to the VDisk.

One advantage offered by this virtualization is to insulate applications and host systems from changes in the underlying storage environment. Data, at the VDisk or LUN level, can be transparently moved to a different storage device or different storage tier without the application knowing. For more information about the SVC, refer to: *IBM System Storage SAN Volume Controller*, SG24-6423, which is available for download at:

<http://www.redbooks.ibm.com/abstracts/sg246423.html?Open>

5.7.2 IBM Virtualization Engine TS7510

The IBM Virtualization Engine TS7510 is a virtual tape library. The TS7510 combines hardware and software into an integrated solution designed to provide tape virtualization for open systems servers connecting over Fibre Channel physical connections.

The TS7510 combines IBM server technology, disk technology and tape technology, and is designed to virtualize, or emulate tape libraries, tape drives, and tape media. Real tape resources can then be attached to the TS7510 to help address Information Lifecycle Management and business continuance. The TS7510 is designed to help customers achieve the following throughput efficiencies:

- ▶ Reduce backup window
- ▶ Improve restore process
- ▶ Facilitate data sharing

The TS7510 is valid ILM solution because it can be used directly by ILM applications that only support tape devices. By redirecting the tape writes and reads to virtual tape you can probably improve time to data and speed up data retrievals. For more information about the TS7510, refer to *IBM Virtualization Engine TS7510: Tape Virtualization for Open Systems Servers*, SG24-7189, that is available for download at:

<http://www.redbooks.ibm.com/abstracts/sg247189.html?Open>

IBM System Storage DR550

IBM System Storage DR550 and DR550 Express systems are designed as pre-configured offerings with servers, storage, and software integrated. The offerings help to preserve and retain electronic business records, either to comply with government and industry regulations, or simply because there is a business requirement for retaining data.

This chapter presents an overview of the DR550 and the DR550 Express. We describe their core components, unique characteristics, and supported applications.

For additional information, refer to:

- ▶ The IBM System Storage Web site:
<http://www.storage.ibm.com/dr550>
- ▶ The IBM Redbook *Understanding the IBM System Storage DR550*, SG24-7091, which is available for download at:
<http://www.redbooks.ibm.com/abstracts/sg247091.html?Open>

We have also included a short overview of available services offerings related to DR550 in Appendix A, “DR550 services offerings” on page 295.

6.1 DR550 data retention solutions

System Storage DR550 and DR550 Express offer scalable data retention solutions to store, retrieve, manage, share, and protect regulated and non-regulated data. DR550 offers secure archival and retention, tiered storage support, synchronous and asynchronous replication capabilities (also known as Metro Mirror and Global Mirror) to help organizations address emerging government and industry-regulatory requirements and corporate governance practices. It is well-suited for archiving e-mail, digital images, database applications, instant messages, account records, contracts or insurance claim documents, and a range of other data.

The DR550 offerings:

- ▶ Provide pre-configured, integrated hardware and software solutions to store, retrieve, manage, share, and protect regulated and non-regulated data.
- ▶ Offer advanced data protection options such as encryption and policy enforcement.
- ▶ Offer a broad suite of software features for policy- and event-based data management.
- ▶ Provide optional encryption for data on its physical disk and attached storage devices (for example, tape).
- ▶ Offer automatic provisioning, migration, expiration, and archiving capabilities.
- ▶ Provide the ability to use advanced WORM tape to back up data objects.
- ▶ Provide a high-availability option designed to avoid single points of failure.
- ▶ Provide optional synchronous and asynchronous data replication between local and remote sites.

The IBM System Storage DR550 and DR550 Express solutions integrate a range of technologies as pre-configured solutions. These solutions provide upgrade options for connectivity and storage capacity, helping to manage up to 89.6 TB of physical disk storage capacity, and additional external tape or optical storage to petabytes of storage per system. These solutions support the ability to retain data without alteration throughout their designated retention period.

6.1.1 IBM System Storage DR550

IBM System Storage DR550, one of IBM Data Retention offerings, is an integrated offering for clients that have to retain and preserve electronic business records. The DR550 packages storage, server, and software retention components into a lockable cabinet. Integrating IBM System P5 servers (using POWER5™ processors) with IBM System Storage and TotalStorage products and IBM System Storage Archive Manager software, this system is designed to provide a central point of control to help manage growing compliance and data retention requirements.

The powerful system, which fits into a lockable cabinet, supports the ability to retain data and helps prevent tampering alteration. The system's compact design can help with fast and easy deployment, and incorporates an open and flexible architecture. The DR550 can be shipped with a minimum of 5.6 terabytes of physical capacity and can expand up to 89.6 terabytes.

Figure 6-1 shows a DR550 configuration with dual server and 44.8 TB physical disk storage.



Figure 6-1 IBM System Storage DR550

Technology

At the heart of the offering is IBM System Storage Archive Manager. This new industry changing software is designed to help customers protect the integrity of data as well as to automatically enforce data retention policies. Using policy-based management, data can be stored indefinitely, can be expired based on a retention event, or have a predetermined expiration date. In addition, the retention enforcement feature can be applied to data using deletion hold and release interfaces which hold data for an indefinite period of time, regardless of the expiration date or defined event.

The policy software is also designed to prevent modifications or deletions after the data is stored. With support for open standards, the new technology is designed to provide customers flexibility to use a variety of content management or archive applications. The System Storage Archive Manager is embedded on an IBM System P5 520 using POWER5+™ processors. This entry-level server has many of the attributes of IBM high-end servers, representing outstanding technology advancements.

Tape storage can be critical for long-term data archiving, and IBM provides customers with a comprehensive range of tape solutions. The IBM System Storage DR550 supports IBM TotalStorage Enterprise Tape Drive 3592, IBM System Storage TS1120 drive, and the IBM Linear Tape Open family of tape products. Write Once Read Many (WORM) cartridges are recommended due to the permanent nature of data stored with the DR550.

We strongly recommend that the 3592 with WORM cartridges be used to take advantage of tape media encoded to enforce nonrewrite and non-erase capability. This complementary capability will be of particular interest to customers that have to store large quantities of electronic records to meet regulatory and internal audit requirements. The DR550 is available in two basic configurations: single node (one POWER5+ server) and dual node (two clustered POWER5+ servers).

Hardware overview

The DR550 includes one or two IBM System P5 520 servers running AIX 5.3. When configured with two 520 servers, the servers are set up in an HACMP™ 5.3 configuration. Both P5 520s have the same hardware configuration. When configured with one 520 server, no HACMP software is included.

IBM System P5 520

The IBM System P5 520 (referred to hereafter as the P5 520 when discussing the DR550) is a cost effective, high performance, space-efficient server that uses advanced IBM technology. The P5 520 uses the POWER5+ microprocessor, and is designed for use in LAN clustered environments. The P5 520 is a member of the symmetric multiprocessing (SMP) UNIX servers from IBM. The P5 520 (product number 9131-52A) is a 4-EIA (4U), 19-inch rack-mounted server. The P5 520 is configured as a 2-core system with 1.9 GHz processors. The total system memory installed is 1024 MB.

The P5 520 includes six hot-plug PCI-X slots, an integrated dual channel Ultra320 SCSI controller, two 10/100/1000 Mbps integrated Ethernet controllers, and eight front-accessible disk bays supporting hot-swappable disks (two are populated with 36.4 GB Ultra3 10K RPM disk drives). These disk bays are designed to provide high system availability and growth by allowing the removal or addition of disk drives without disrupting service. The internal disk storage is configured as mirrored disk for high availability. Figure 6-2 shows the front view of a P5 520 server.



Figure 6-2 Front view of P5 520 server

In addition to the disk drives, there are also three media bays available:

- ▶ Media - dev0 - not used for DR550
- ▶ Media - dev1 - Slimline DVD-RAM (FC 1993)
- ▶ SCSI tape drive (not included)

On the back of the server, different ports and slots are included.

Figure 6-3 shows the back of a P5 520 server.

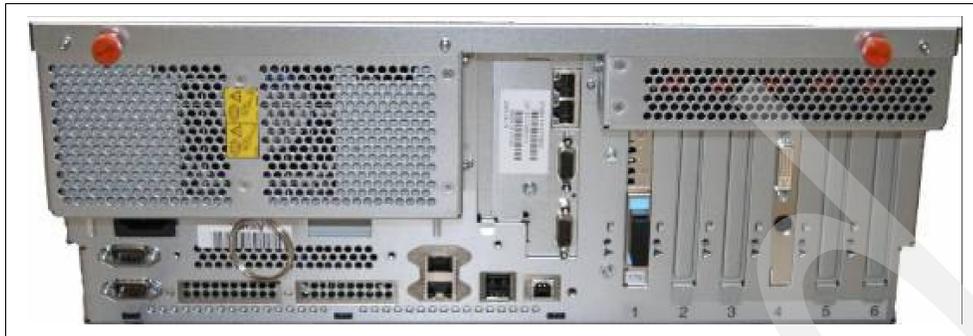


Figure 6-3 Back view of P5 520 server

The ports and slots included are:

► PCI-X slots:

The P5 520 provides multiple hot-plug PCI-X slots. The number and type of adapters installed is dependent on the configuration selected. The following adapters are installed:

- Three 2 Gigabit Fibre Channel PCI-X adapters (two for connections to the internal SAN for disk attachment and one for connection to the internal SAN for tape attachment) (FC 5716) - located in slots 1, 4, 5.
- One 10/100/1000 Mbps dual port Ethernet PCI adapter II (FC 1983 - TX version or FC 1984 - SX version) - located in slot 3 and used for connection to the client network.
- One POWER™ GXT135P Graphics Accelerator with Digital support adapter (FC 1980) - located in slot 2.

► I/O ports:

The P5 520 includes several native I/O ports as part of the basic configuration:

- Two 10/100/1000 Ethernet ports (for copper based connections). Both are used for connections to the DS4100 and used for management purposes only (no changes should be made in these connections).
- Two serial ports (RS232). These are not used with DR550.
- Two USB ports. One of these is used to connect to the keyboard and mouse - the other port is not used.
- Two RIO ports. These are not used by DR550
- Two HMC (Hardware Management Console) ports. One is used for connection to the HMC server in the rack.
- 2 SPCN ports. These are not used by DR550.

The Converged Service Processor² (CSP) is on a dedicated card plugged into the main system planar, which is designed to continuously monitor system operations, taking preventive or corrective actions to promote quick problem resolution and high system availability.

Additional features are designed into pSeries® servers to provide an extensive set of reliability, availability, and serviceability (RAS) features such as improved fault isolation, recovery from errors without stopping the system, avoidance of recurring failures, and predictive failure analysis.

Management Console

Included in the DR550 is a set of integrated management components. This includes the Hardware Management Console (HMC) as well as a flat panel monitor, keyboard and mouse. The HMC (7310-CR3) is a dedicated rack-mounted workstation that allows the user to configure and manage call home support. The HMC has other capabilities (partitioning, Capacity on Demand) that are not used in the DR550. The HMC includes the management application used to set up call home.

To help ensure console functionality, the HMC is not available as a general purpose computing resource. The HMC offers a service focal point for the 520 servers that are attached. It is connected to a dedicated port on the service processor of the POWER5 system via an Ethernet connection. Tools are included for problem determination and service support, such as call-home and error log notification, through the internet or via modem. The customer must supply the connection to the network or phone system. The HMC is connected to the keyboard, mouse and monitor installed in the rack.

The IBM 7316-TF3 is a rack-mounted flat panel console kit consisting of a 17 inch (337.9 mm x 270.3 mm) flat panel color monitor, rack keyboard tray, IBM travel keyboard (English only), and the Netbay LCM switch. This is packaged as a 1U kit and is mounted in the rack along with the other DR550 components. The Netbay LCM Switch is mounted in the same rack space, located behind the flat panel monitor. The IBM Travel Keyboard is configured for English. An integrated "mouse" is included in the keyboard. The HMC and the P5 520 servers are connected to the Netbay LCM switch so that the monitor and keyboard can access all three servers.

IBM TotalStorage DS4700 and TotalStorage DS4000 EXP810

The DR550 includes one or two IBM TotalStorage DS4700 Midrange Disk System (hereafter referred to as the DS4700) depending on capacity. The disk capacity used by the DS4700s is provided by the IBM TotalStorage EXP810 (hereafter referred to as the EXP810).

The DS4700 is an affordable, scalable storage server for clustering applications such as the Data Retention application. Its modular architecture -which includes Dynamic Capacity Expansion and Dynamic Volume Expansion-is designed to support e-business on demand® environments by helping to enable storage to grow as demands increase. Autonomic features such as online firmware upgrades also help enhance the system's usability.

The single server capacity is 8 and 16 Terabytes. The dual server comes in capacities of 8,16, 32, 56 and 112 terabytes. The DS4700 is designed to allow upgrades while keeping data intact, helping to minimize disruptions during upgrades. It also supports online controller firmware upgrades, to help provide high performance and functionality. Events such as upgrades to support the latest version of DS4000 Storage Manager can also often be executed without stopping operations.

Storage controller features

The following features are included:

- ▶ Storage controller: One or two IBM System Storage DS4700s (depends on capacity)
- ▶ Maximum of 14 IBM TotalStorage DS4000 EXP810 units, each with 500GB SATA hard disk drives
- ▶ Optional Metro Mirror or Global Mirror for replication and 2005-B16 FC switches for the DR550 with Mirroring option
- ▶ Fibre Channel switch — IBM 2005-B16
- ▶ IBM 7014 rack model T00:
 - Rack security feature
 - Additional power distribution units (PDUs)

IBM TotalStorage SAN Switch

Two IBM TotalStorage SAN Fibre Channel Switches are used to interconnect both P5 520 servers with the DS4700s to create a SAN (dual node configurations). Tape attachment such as the 3592, TS1120 or LTO can be done using the additional ports on the switches. The switches (2005-B16; see Figure 6-4) build two independent SANs, which are designed to be fully redundant for high availability. This implementation in the DR550 is designed to provide high performance, scalability, and high fault tolerance.



Figure 6-4 2005-B16

For the single node configurations, only one switch (2005-B16) is included. This creates a single independent SAN and can be used for both disk and tape access. The 2005-B16 is a 16-port, dual speed, auto-sensing Fibre Channel switch. Eight ports are populated with 2 gigabit shortwave transceivers when the DR550 is configured for single copy mode. Twelve ports are populated with 2 gigabit short wave transceivers when the DR550 is configured for enhanced remote volume mirroring. This dual implementation is designed to provide a fault tolerant fabric topology, to help avoid single points of failure.

Accessing the switches

If you have to access the switches to review the zoning information, error messages, or other information, you must connect Ethernet cables (provided by the customer) to the Ethernet port on the switch. These cables should also be connected to the customer network. You can then access the switch using the IP address. The User ID is ADMIN and the password is PASSWORD. You should change this password to conform with site security guidelines.

If you have to review the configuration or zoning within the switches, the IP address for switch 1 is 192.168.1.31 and switch 2 (only installed in dual node configurations) is 192.168.1.32.

These addresses should not be changed. To gain access to the switches via the IP network, you must provide Ethernet cables and ports on your existing Ethernet network. After the connections have been made, then you can connect to the IP address and use the management tools provided by the switch.

Should one of the switches fail (dual node configurations only), the logical volumes within the DS4700 systems are available through the other controller and switch.

Software overview

The DR550 consists of hardware and software components. In this section, we describe the software components.

High Availability Cluster Multi-Processing (HACMP) for AIX

The data retention application can be a business critical application. The DR550 can provide a high availability environment by leveraging the capabilities of AIX and High Availability Cluster Multi-Processing (HACMP) with dual P5 servers and redundant networks. This is referred to as the dual node configuration. IBM also offers a single node configuration that does not include HACMP.

HACMP is designed to maintain operational applications such as System Storage Archive Manager if a component in a cluster node fails. In case of a component failure, HACMP is designed to move the application along with the resources from the active node to the standby (passive) node in the DR550.

Cluster nodes

The two P5 520 servers running AIX with HACMP daemons are Server nodes that share resources—disks, volume groups, file systems, networks, and network IP addresses. In this HACMP cluster, the two cluster nodes communicate with each other over a private Ethernet IP network. If one of the network interface cards fails, HACMP is designed to preserve communication by transferring the traffic to another physical network interface card on the same node. If a “connection” to the node fails, HACMP is designed to transfer resources to the backup node to which it has access.

In addition, heartbeats are sent between the nodes over the cluster networks to check on the health of the other cluster node. If the passive standby node detects no heartbeats from the active node, the active node is considered as failed and HACMP is designed to automatically transfer resources to the passive standby node.

Within the DR550 (dual node configuration only), HACMP is configured as follows:

- ▶ The clusters are set up in Hot Standby (active/passive) mode.
- ▶ The resource groups are set up in cascading mode.
- ▶ The volume group is set up in enhanced concurrent mode.

System Storage Archive Manager

IBM System Storage Archive Manager (this is the new name for IBM Tivoli Storage Manager for Data Retention) is designed provide archive services and to prevent critical data from being erased or rewritten. This software can help address requirements defined by many regulatory agencies for retention and disposition of data. Key features include these:

- ▶ **Data retention protection:** This feature is designed to prevent deliberate or accidental deletion of data until its specified retention criterion is met.
- ▶ **Event-based retention policy:** In some cases, retention must be based on an external event such as closing a brokerage account. System Storage Archive Manager supports event-based retention policy to allow data retention to be based on an event other than the storage of the data. This feature must be enabled via the commands sent by the content management application.
- ▶ **Deletion hold:** In order to ensure that records are not deleted when a regulatory retention period has lapsed but other requirements mandate that the records continue to be maintained, System Storage Archive Manager includes deletion hold. Using this feature will help prevent stored data from being deleted until the hold is released. This feature must be enabled via the commands sent by the content management application.
- ▶ **Data encryption:** 128-bit Advanced Encryption Standard (AES) is now available for the Archive API Client. Data can now be encrypted before transmitting to the DR550 and would then be stored on the disk/tape in an encrypted format.

For more information about System Storage Archive Manager, refer to 4.1, “Tivoli Storage Manager concepts” on page 74.

System Storage Archive Manager API Client

The System Storage Archive Manager API Client is used, in conjunction with System Storage Archive Manager server code, as the link to applications that produce or manage information to be stored, retrieved and retained. Content management applications, such as the IBM DB2 Content Manager, identify information to be retained.

The content management application calls the System Storage Archive Manager (SSAM) archive API Client to store, retrieve, and communicate retention criteria to the SSAM server. The SSAM API Client must be installed on the application or middleware server that is used to initiate requests to DR550. Then, the application or middleware server must call the SSAM API to initiate a task within the DR550. Some applications and middleware include the API client as part of their code. Others require it to be installed separately.

DS4000 Storage Manager

The DS4000 Storage Manager Version software used (hereafter referred to as Storage Manager) is only available as part of the DR550 and is not available for download from the Web. This version has been enhanced to provide additional protection.

Storage Manager is designed to support centralized management of the DS4700s in the DR550. Storage Manager is designed to allow administrators to quickly configure and monitor storage from a Java-based GUI interface. It is also designed to allow them to customize and change settings as well as configure new volumes, define mappings, handle routine maintenance, and dynamically add new enclosures and capacity to existing volumes, without interrupting user access to data. Failover drivers, performance-tuning routines, and cluster support are also standard features of Storage Manager.

Using the DS4000 Storage Manager, the DS4700 is partitioned into a single partition at the factory. The P5 520 servers are connected to the DS4700s via Ethernet cables. This connection is used to manage the DS4000. For the single node configuration, DS4000 Storage Manager runs in the P5 520 server. For the dual node configuration, DS4000 Storage Manager runs in both servers. Server #2 is used to manage DS4700 #1 and Server #1 is used to DS4700 #2 (if present in the configuration).

Attention: Only this special version of DS4000 Storage Manager should be used with the DR550. You should not use this version with other DS4000 or FAStT disk systems, and you should not replace this version with a standard version of DS4000 Storage Manager (even if a newer version is available).

6.1.2 IBM System Storage DR550 Express

IBM System Storage DR550 Express is an integrated data retention offering for clients that have to retain and preserve electronic business records. The DR550 Express packages storage, server, and software retention components into a pre-configured offering.

Integrating IBM eServer™ pSeries POWER5 processor-based servers and IBM System Storage Archive Manager software, this offering provides, like the DR550, a central point of control to help manage growing compliance and data retention requirements. The system is designed to be mounted into a standard 19 inch rack. A lockable rack for added security can be purchased separately if required, as the cabinet is not included with DR550 Express.

The system supports the ability to retain data and inhibit tampering or alteration. The system's compact design can help with fast and easy deployment, and incorporates an open and flexible architecture. Figure 6-5 shows the base configuration of the DR550 Express with 1.1 TB of physical disk storage.



Figure 6-5 IBM System Storage DR550 Express

The DR550 Express is shipped with approximately 1 TB of physical capacity and can be expanded to 5.1 TB or 9.1 TB physical capacity.

Tip: Consider carefully the decision to opt for a DR550 Express solution. Maximum physical disk storage capacity for the DR550 Express is currently 9.1 TB. A DR550 Express cannot be upgraded to DR550. It can be replaced by a DR550, but additional migration services are required then.

Technology

The DR550 Express is based on the same core software as the DR550, that is the IBM System Storage Archive Manager. System Storage Archive Manager is installed in the IBM eServer P5 520 using POWER5 processor. The IBM System Storage DR550 Express supports IBM TotalStorage Enterprise Tape Drive 3592, System Storage TS1120 as well as the IBM Linear Tape Open family of tape products (using Write Once Read Many or WORM cartridges). The tape drives can be installed in tape libraries such as the IBM 3494 (3592 and TS1120 drives), 3581 (with LTO Gen 3 drive), 3582 (with LTO Gen 3 drives), 3583 (with LTO Gen 3 drives), 3584 (with LTO Gen 3, 3592 and/or TS3310 drives) or the IBM System Storage TS3310 (with LTO Gen 3 drives).

Other tape drives and libraries are supported as well. Due to the permanent nature of data stored with the DR550 Express, we strongly recommend that the tape drives always use WORM cartridges to take advantage of tape media encoded to enforce non-rewrite and non-erase capability. This complementary capability will be of particular interest to customers that have to store large quantities of electronic records to meet regulatory and internal information retention requirements. The DR550 Express is pre-configured to support both disk and tape storage.

Hardware overview

The DR550 Express includes one IBM eServer P5 520 server running AIX 5.3, a flat panel monitor and keyboard, and a Fibre Channel SAN Switch. No clustering option is available.

IBM eServer P5 520

The IBM eServer POWER5 520 (referred to hereafter as the P5 520 when discussing the DR550 Express) is a cost-effective, high performance, space-efficient server that uses advanced IBM technology. The P5 520 uses the POWER5 microprocessor, and is designed for use in LAN clustered environments. The P5 520 is a member of the symmetric multiprocessing (SMP) Unix servers from IBM. The P5 520 (product number 9111-520) is a 4-EIA (4U), 19-inch rack-mounted server (you will have to provide space in an existing rack or purchase a new rack separately). The P5 520 is configured with a 1-way 1.5 GHz processor. The total system memory installed is 512 MB.

The P5 520 includes six hot-plug PCI-X slots, an integrated dual channel Ultra320 SCSI controller, two 10/100/1000 Mbps integrated Ethernet controllers, and eight front-accessible disk bays supporting hot-swappable disks (all eight are populated with 146 GB Ultra3 10K RPM disk drives). These disk bays are designed to provide high system availability and growth by allowing the removal or addition of disk drives without disrupting service. The internal disk storage uses RAID-5 protection for high availability.

In addition to the disk drives, there are also three media bays available:

- ▶ Media - dev0 - not used for DR550
- ▶ Media - dev1 - Slimline DVD-RAM (FC 5751)
- ▶ SCSI tape drive (not included)

On the back of the server, the following ports and slots are included:

- ▶ PCI-X slots:
 - One 2 Gigabit Fibre Channel PCI-X adapter (FC5716) for connection to the internal SAN (for tape attachment).
 - One POWER GXT135P Graphics Accelerator with Digital support adapter (FC2849), used to connect to the integrated monitor.
 - Adapters for a network connection, which the customer must choose:
 - For fiber optic connections, select the single Port Gigabit Ethernet-SX PCI-X adapter (FC5701).
 - For copper connections, use the integrated 10/100/1000 Ethernet ports on the p520 server.

- ▶ I/O ports:

The P5 520 includes several native I/O ports as part of the basic configuration:

- Two 10/100/1000 Ethernet ports (for copper based connections). These are used for connection to the external customer network.
- Two serial ports (RS232). These are not used with DR550 Express.
- Two USB ports. One of these is used to connect to the keyboard and mouse.
- Two RIO ports. These are not used by DR550 Express.
- Two HMC (Hardware Management Console) ports. These are not used by DR550 Express.
- Two SPCN ports. These are not used by DR550 Express.

The Converged Service Processor² (CSP) is on a dedicated card plugged into the main system planar, which is designed to continuously monitor system operations, taking preventive or corrective actions to promote quick problem resolution and high system availability. Additional features are designed into pSeries servers to provide an extensive set of reliability, availability, and serviceability (RAS) features such as improved fault isolation, recovery from errors without stopping the system, avoidance of recurring failures, and predictive failure analysis.

Flat Panel Console Kit

The DR550 Express includes an integrated flat panel monitor, keyboard, and mouse. The IBM 7316-TF3 is a rack-mounted flat panel console kit consisting of a 17 inch (337.9 mm x 270.3 mm) flat panel color monitor, rack keyboard tray, IBM travel keyboard (English only), and the Netbay LCM switch. This is packaged as a 1U kit and can be mounted in a customer provided rack along with the other DR550 Express components.

The Netbay LCM Switch is mounted in the same rack space, located behind the flat panel monitor. The IBM Travel Keyboard is configured for English. An integrated “mouse” is included in the keyboard. The POWER5 520 server is connected to the Netbay LCM switch so that the monitor and keyboard can access the server.

IBM TotalStorage SAN Switch

One IBM TotalStorage SAN Fibre Channel Switch is included in the offering. The switch is used to interconnect the P5 520 server with a Fibre Channel based tape solution such as the IBM 3592 or IBM LTO based libraries. The switch (2005-B16) supports multiple connections to the tape. The 2005-B16 is a 16-port, dual speed, auto-sensing Fibre Channel switch. Eight ports are populated with 2 gigabit shortwave transceivers.

Accessing the switch

If you have to access the switch to review the zoning information, error messages, or other information, you must connect Ethernet cables (which are provided by the customer) to the Ethernet port on the switch. These cables would also have to be connected to the customer network. You can then access the switch using the IP address. The userid is ADMIN and the password is PASSWORD. You should change this password to confirm with site security guidelines. If you have to review the configuration or zoning within the switches, the IP address for the switch is 192.168.1.31. This address should not be changed. To gain access to the switch via the IP network, you must provide an Ethernet cable and ports on your existing Ethernet network. After the connections have been made, then you can connect to the IP address and use the management tools provided by the switch.

IBM TotalStorage DS4700 Midrange Disk System: Optional

The DR550 Express can include one IBM TotalStorage DS4700 Midrange Disk System (hereafter referred to as the DS4700). In Table 6-1 we list the characteristics of the DS4700 Storage Server inside the DR550 Express.

Table 6-1 IBM DS4100 Storage Server in the DR550 Express at a glance

Characteristics	Descriptions
Model	1814-70A
RAID controller	Dual active 4 GB RAID controllers
Cache	2048 MB total, battery-backed
Host interface	4 -Fibre Channel (FC) Switched and FC Arbitrated Loop (FC-AL) standard
Drive interface	Redundant 4 Gbps FC-AL connections
EXP100 drives	500 GB 7200 RPM SATA disk drives
RAID	Level 5 configured. RAID-10 can be configured at the customer's site using an optional IBM Services consultant
Maximum drives supported	8 or 16 Serial ATA drives
Fans	Dual redundant, hot-swappable
Management software	IBM DS4000 Storage Manager version 9.12.65 (Special version for exclusive use with DR550 Express)

IBM Entry Rack Cabinet Model 7014-S25: Optional

Manufactured to meet the EIA 310-D standard, the IBM 7014 Model S25 Rack accommodates system units and expansion drawers that are designed for 19-inch rack mounting. This rack features the EIA standard square hole design in the mounting rails.

The Model S25 Rack has 25 EIA units of space for mounting system units and expansion drawers designed for 19-inch rack mounting. The overall rack height is 49 inches, and the distance between the front and back EIA mounting rails is 28.3 inches.

The rack comes with removable side covers and locking front and rear doors. The front door is reversible so that it can be configured for either left or right opening. The rear door is split vertically in the middle and hinges on both the left and right sides. The rack is available in IBM black. Filler panels in the front of the rack, behind the door, cover all EIA space that is not filled with rack mounted devices.

The rack will be shipped pre-assembled with the DR550 Express components (server and monitor) if ordered at the same time. The rack ships with a front stabilizer bracket for installing or servicing system units from the front of the rack. The rack is mounted on casters, two swivel casters in front and two fixed casters in the back, that support easy movement of the rack and rack contents. An adjustable foot near each caster can be lowered to the floor to restrict unwanted motion.

Software overview

The DR550 Express comes with System Storage Archive Manager and DS4000 Storage Manager Version 9.12.65 pre-installed. See the “Software overview” on page 147 for additional information. Because the DR550 Express ships only as a single server solution, no cluster software is installed.

Attention: Only this special version of DS4000 Storage Manager should be used with the DR550 Express. You should not use this version with other DS4000 or FASTT disk systems, and you should not replace this version with a standard version of DS4000 Storage Manager (even if a newer version is available).

6.2 DR550 functions and capabilities

In this section, we describe the unique features and functions of the DR550 that differentiate this solution from other data retention solutions in the market.

6.2.1 Flexible retention policies

DR550 provides the functionality to:

- ▶ Enable management of data that has no explicit retention period, such as employee data (as long as employed) and customer data (as long as an account is open), through an event-based records management feature. It is also an excellent feature for documents that have a specific retention period that can be terminated early (for example, mortgages, or financial time deposits), or for those documents that have no specific retention period (for example, insurance policies). It can help protect these records from deletion until a specific event occurs.
- ▶ Allow a designated object or group of objects to be protected against the normal end of life (policy expiration) process by using a deletion hold management feature. This can be very useful in the event that a record or set of records has to be retained for legal, audit, or other reasons.

- ▶ Help protect data by preventing explicit data deletion before retention criteria expiration.
- ▶ Enforce data-protection policies that maintain the data in non-erasable and non-rewriteable formats.
- ▶ Permit users to automatically archive files from their workstations or file servers to data-retention protected storage, and to retrieve archived copies of files to their local workstations or file servers through an archive client.

6.2.2 Tiered storage solution and scalability

The DR550 enables data management on multiple tiers of storage (for example, tape, optical, CD, DVD) using a tiered storage management feature to provide a more cost-effective solution:

- ▶ Almost unlimited secondary storage (such as tape)
- ▶ Disk cache limited upgrade options
- ▶ Tape attachment, LTO or 3592

6.2.3 Data migration capabilities

The DR550 offers the ability to migrate to different storage technologies, which is important for long retention times, where technology advances and technology obsolescence during an extended lifetime require migration.

Disaster protection

DR550 can also help protect customer data during disasters. IBM System Storage DR550 provides support for Metro Mirroring. This new feature allows two real-time synchronized copies of data to be maintained on DR550s in separate locations. DR550 also provides the capability to use tape libraries with IBM TS1120 or LTO Ultrium 3 tape drives to provide efficient and cost-effective replications of the data objects and the DR550 database to support off-site data storage and recovery in the event of a failure or the requirement to relocate to an alternate facility.

6.2.4 Data encryption

Enabling companies to protect their data when transmitted over the network or saved to disk, data encryption can provide enhanced security for businesses via 128 bit AES or 56 bit DES encryption technology. Within Tivoli Storage Manager, encryption has already been a proven technology for years, and now this is also true in System Storage Archive Manager. Encryption options allow DR550 to manage encryption keys (key management for each object) transparent to the application, or allow an application to manage encryption keys externally to DR550. The application stores and uses the keys to retrieve. Encryption is enabled or disabled through an option in the client.

6.2.5 Performance

DR550 can offer excellent performance, especially when processing sessions with more than a single object. Planning for the appropriate DR550 configuration should be done as part of the overall project. The DR550 has been tuned to provide balanced performance for both small and larger disk capacities. The number of DR550s required should be based on the number of objects to be archived and retrieved and the ability of the content management application to support multiple objects per session.

The IBM System Storage DR550 Performance Measurements document provides performance and capacity planning information for the DR550. The paper provides both measurements in the form of megabytes/second and objects/second. In addition, the paper provides a detailed configuration list. It can be found on our Web page at:

<http://www.storage.ibm.com/disk/dr/performance.html>

6.3 ISV support list

For the DR550 and DR550 Express to function within a customer IT environment, information appropriate to be retained must be identified and supplied to the DR550. This can be accomplished with a content management application, which provides information to the DR550 or DR550 Express via the System Storage Archive Manager API client.

6.3.1 IBM DB2 Content Manager

IBM DB2 Content Manager provides a foundation for managing, accessing, and integrating critical business information about demand. It lets you integrate all forms of content, such as document, Web, image, rich media, across diverse business processes and applications, including Siebel, PeopleSoft, and SAP. Content Manager integrates with existing hardware and software investments, both IBM and non-IBM, enabling customers to leverage common infrastructure, achieve a lower cost of ownership, and deliver new, powerful information and services to customers, partners, and employees where and when required. It is composed of two core repository products that are integrated with System Storage Archive Manager for storage of documents into the DR550 or DR550 Express:

- ▶ DB2 Content Manager is optimized for large collections of large objects. It provides imaging, digital asset management, and Web content management. When combined with DB2 Records Manager, it also provides a robust records retention repository for managing the retention of all enterprise documents.
- ▶ DB2 Content Manager OnDemand is optimized to manage very large collections of smaller objects such as statements and checks. It provides output and report management.

There are a number of applications that work with IBM Content Manager to deliver specific solutions. These applications are designed to use Content Manager functions and can send data to be stored in DR550 or DR550 Express:

- ▶ IBM CommonStore for Exchange Server
- ▶ IBM CommonStore for Lotus Domino
- ▶ IBM CommonStore for SAP
- ▶ BRMS (iSeries) (also via IFS to BRMS)

More information about the DB2 Content Manager portfolio of products can be found in Chapter 3, "Information Management software" on page 43.

6.3.2 SSAM archive client

System Storage Archive Manager comes with client archive software enabling users to archive and retrieve directly from or to their workstations or file servers to protected storage.

6.3.3 Other content management applications

Consult your application software vendor to determine if your applications support the DR550 API. A number of application providers have enhanced their software to include this support. The current list includes:

- ▶ AXS-One
- ▶ BrainTribe (formerly Compendium)
- ▶ Caminosoft
- ▶ Ceyoniq
- ▶ Easy Software
- ▶ FileNet
- ▶ Hummingbird
- ▶ Hyland Software (OnBase)
- ▶ Hyperwave
- ▶ IRIS Software (Documentum Connector)
- ▶ MBS Technologies (iSeries Connector for IBM CM V5)
- ▶ OpenText (formerly IXOS)
- ▶ Princeton Softech Active Archive Solution for PeopleSoft; for Siebel; for Oracle
- ▶ Saperion
- ▶ SER Solutions
- ▶ Symantec Enterprise Vault (formerly KVS)
- ▶ Waters (Creon Labs, NuGenesis)
- ▶ Windream
- ▶ Zantaz

Only applications or middleware using the API can send data to DR550. Information regarding the System Storage Archive Manager API Client might be found at:

<http://publib.boulder.ibm.com/infocenter/tivihelp/index.jsp?toc=/com.ibm.itstorage.doc/toc.xml>

For additional information about qualified ISVs, refer to the interoperability Web page at:

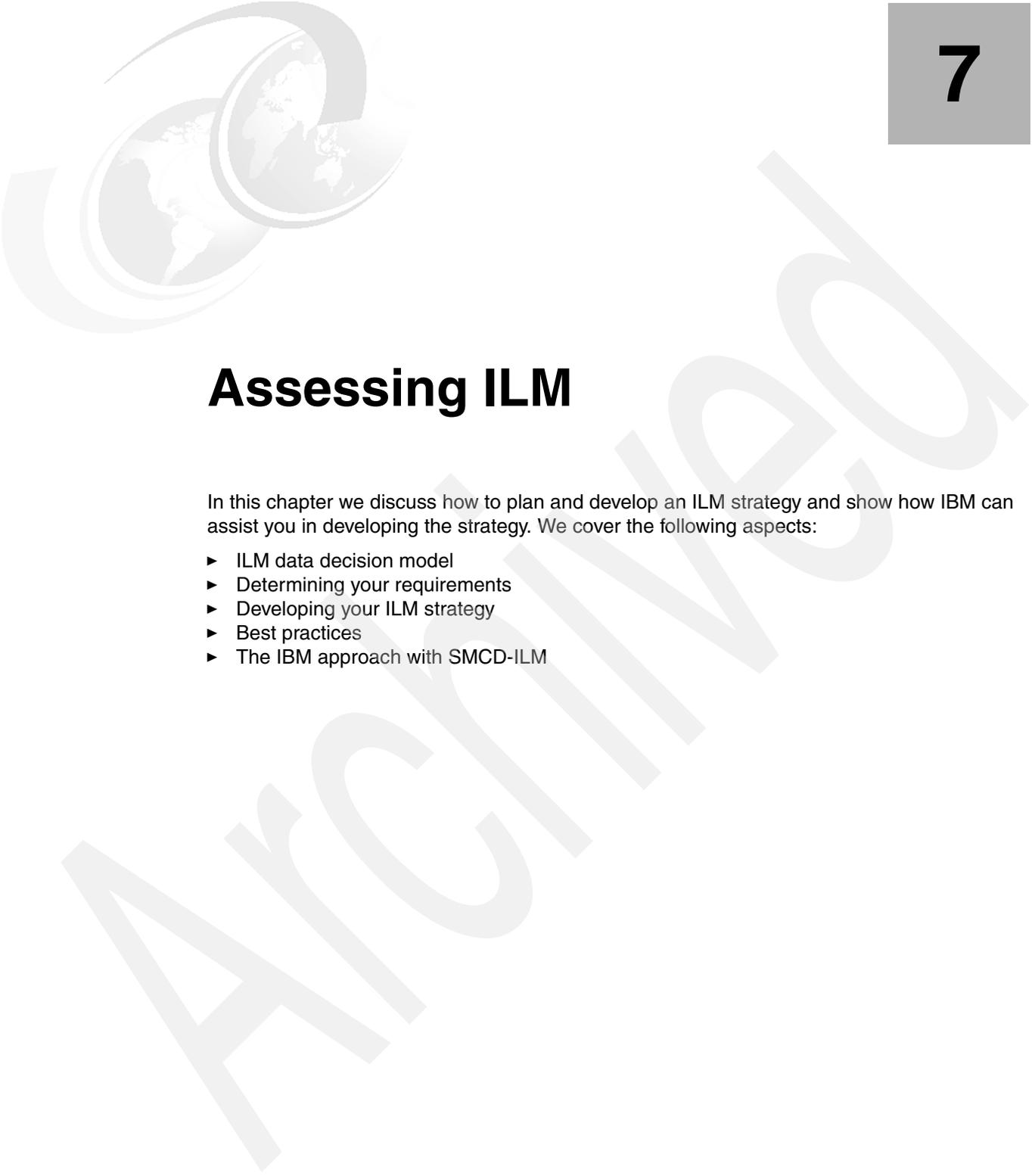
<http://www.storage.ibm.com/dr550>

Strategies and solutions

In this part of the book we discuss the following topics:

- ▶ Assessing ILM, which includes developing an ILM strategy
- ▶ Content Management and integrated Storage Management
- ▶ File system archiving and retention, including a description of file systems and their relationship with ILM practices and retention management of data
- ▶ Other archiving solutions

Archived



Assessing ILM

In this chapter we discuss how to plan and develop an ILM strategy and show how IBM can assist you in developing the strategy. We cover the following aspects:

- ▶ ILM data decision model
- ▶ Determining your requirements
- ▶ Developing your ILM strategy
- ▶ Best practices
- ▶ The IBM approach with SMCD-ILM

7.1 An ILM decision model

In “ILM six best practices” on page 34 we discussed some possible ILM approaches, classified into six best practices, which span both technological and organizational aspects. Now we outline and illustrate a decision model related to the technological aspects. This model concentrates on possible technological solutions for a set of different application environments, with different ILM requirements.

We show tiered storage and ILM solutions for various kinds of environments. We can make a distinction between two broad kinds of applications or environments: database type environments and file system oriented environments. There are different problems and different types of solutions for each one, therefore, we approach the two environments separately.

Database data solution selection model

Figure 7-1 shows a solution selection flowchart that can help you identify the best solution for a given database environment.

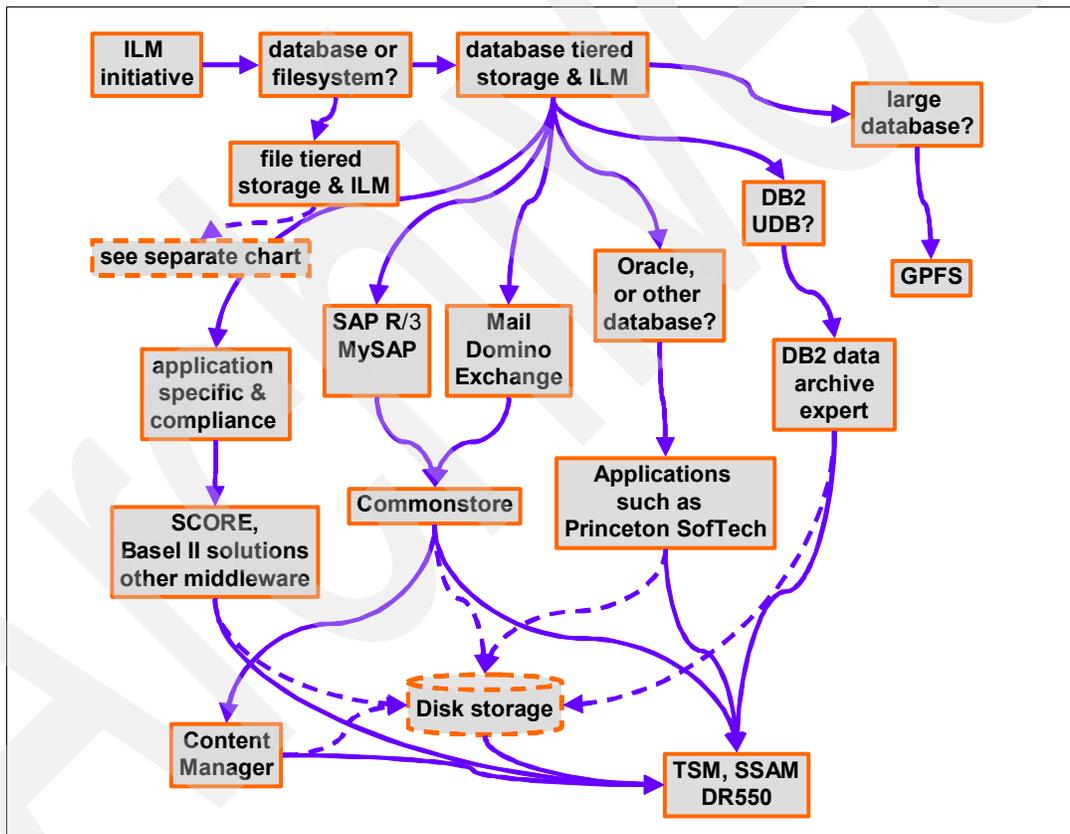


Figure 7-1 A solution selection map for database environments

Starting at the top left, the diagram entry point is ILM initiative, where you are searching for ILM and a tiered storage for your applications. This assumes that you have knowledge about the various applications and environments. We select one specific application and start asking the question: is it a file based or a database type of application?

Examples of file based applications are, rather predictably, file servers, printing output spools, image repositories, and so on. The common theme is that the data resides in some kind of file system, and in general, one individual data object is represented by one single file.

Examples of these database based applications are various database management systems such as DB2, Oracle, MySQL, and Sybase, as well as e-mail applications such as Lotus Domino and Microsoft Exchange. Many other applications fall into this category, applications that have their own proprietary databases.

The diagram in Figure 7-1 on page 160 shows that we have a database type application. For file applications, refer to Figure 7-2 on page 162. Proceeding to the box called *database tiered storage and ILM*, you now have to choose the database type you would like to apply the ILM techniques to. Depending on the type of database and application, there are specific solutions that can be applied. We outline some of the possible solutions here:

- ▶ For large database environments, databases that range in the tens or hundreds of terabytes, you can use a specialized solution called General Parallel File System (GPFS). GPFS is a high-performance shared-disk file system that can provide fast, reliable data access from all nodes in a homogenous or heterogeneous cluster of IBM UNIX servers running either the AIX 5L™ or the Linux operating system.

GPFS allows parallel applications simultaneous access to a set of files (even a single file) from any node that has the GPFS file system mounted, while providing a high level of control over all file system operations. GPFS provides high-performance I/O by “striping” blocks of data from individual files across multiple disks (on multiple storage devices) and reading/writing these blocks in parallel. In addition, GPFS can read or write large blocks of data in a single I/O operation, thereby minimizing overhead. For more information, refer to:

<http://www-03.ibm.com/servers/eserver/clusters/software/gpfs.html>

- ▶ When you have DB2 UDB environments, you can use DB2 Data Archive Expert, part of the IBM DB2 Toolkit for Multiplatforms product. It is a comprehensive data archiving tool that enables you to move seldom-used data to a less costly storage mediums, without any programming. Using this tool, you can save storage space and associated costs, while improving the performance of your DB2 environment. For more information, refer to:

<http://www-306.ibm.com/software/data/db2imstools/db2tools/db2archiveexpert.html>

- ▶ Oracle and other databases can benefit from archiving or decommissioning old data. Princeton Softech Optim can help because it offers a business policy-driven framework to define, extract, access and restore related sets of data from cross-platform and cross-application relational databases. This allows you to control database growth by removing data selectively; separate critical on-line production data from *active reference data*; and research, analyze, and restore *active reference data* selectively. For more information, refer to:

<http://www.princetonsoftech.com/>

- ▶ E-mail systems such as Lotus Domino and Microsoft Exchange tend to grow. Solutions such as DB2 CommonStore for Lotus Domino and DB2 CommonStore for Microsoft Exchange manage e-mail archiving and retrieval for mail databases. Data can be off-loaded to a less expensive storage tier. For more information, refer to:

<http://www-306.ibm.com/software/data/commonstore/>

- ▶ SAP environments can use DB2 CommonStore for SAP, this can help you off-load operational SAP databases, work with non-SAP documents from within SAP Business Objects, and process business documents that reside in an external archiving system. CommonStore for SAP is a middleware server between the SAP ArchiveLink interface and a required back-end archive product such as DB2 Content Manager or Tivoli Storage Manager. For more information refer to:

<http://www-306.ibm.com/software/data/commonstore/sap/>

- ▶ There are many other solutions and service offerings for application specific and compliance environments. One example is SCORE Solution for Compliance in Regulated

Environments (SCORE) that provides a document management solution with application integration, business process management and collaboration functions. We will discuss more applications for specific environments in subsequent sections of this book.

All the data extraction and archival applications and solutions we have illustrated extract data from a database application, which probably resides on high end disk storage devices, and stores the resulting data on less expensive devices. Different applications support different destinations for the resulting data, as schematically illustrated in the diagram in Figure 7-1 on page 160, destinations such as:

- ▶ Content Manager
- ▶ Disk storage
- ▶ Tivoli Storage Manager or a DR550 solution

The type of destination must be evaluated for each application type, for example, data extracted by DB2 CommonStore could be stored using DB2 Content Manager, and DB2 Content Manager writes the data to disk storage and later moves it to a DR550. There is a great flexibility in using multiple storage devices and middleware.

File data solution selection model

A similar type of schematic solution diagram can also be applied to file type data, which resides on a file system. A definition of a file is: a collection of data that is stored on a storage device such as disk and that is manipulated as a single unit with its name. A file can either represent a single object, for example, a document, a group of objects such as a UNIX TAR file, or be part of a larger object, such as a GIF image file that is part of a Web page. File ILM solutions, such as HSM, manage the files on an individual file basis; files are not aggregated into groups. Figure 7-2 shows a possible solution selection methodology for file based application environments.

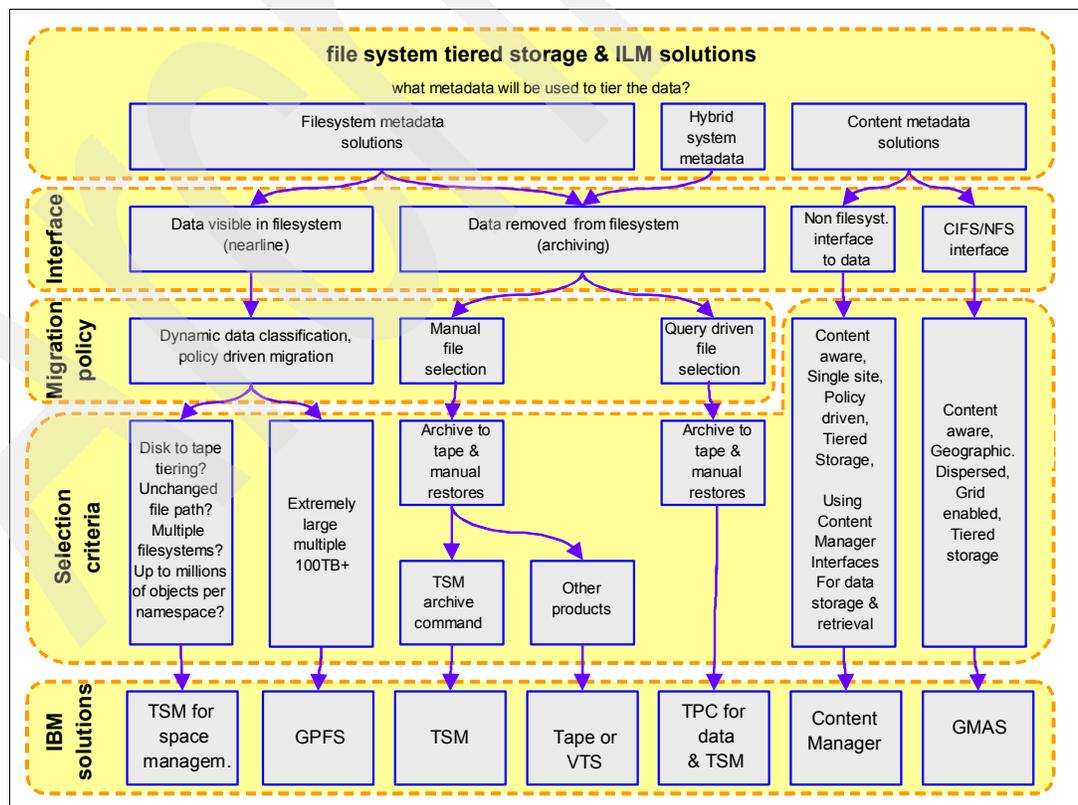


Figure 7-2 A solution selection map for files

The top of the diagram shows the entry point to the solution selection process for file system type data. The first question is, what metadata will be used to tier the data? When you create or update a file in a file system, that file has a set of attributes associated with it, such as:

- ▶ Name
- ▶ File extension, in Windows environments
- ▶ Creation date
- ▶ Last reference date
- ▶ Size
- ▶ Owner and access permissions

These attributes are the *file metadata* — they are information about a set of data, the set of bytes that compose the file. Based on this metadata we can make informed decisions about the importance of the data. For example, we could say that files with extension *.html (World Wide Web browser files) should be stored on intermediate performance media. We could also decide that files that have not been referenced in the last five years can be deleted.

We can also define another, different, kind of metadata: *content metadata*. This is not part of the normal file system metadata, but rather relates to the content of the file. Examples are a search engine that indexes the content of the file's recurring words or an image recognition program that can classify pictures. This kind of metadata requires separate programs, first to explore and extract the file contents, and then to keep track of the attributes it has found. In the following discussion, we give two examples to clarify possible uses of content metadata:

- ▶ A first example is based on the Grid Medical Archive Solution (GMAS). It refers to a positive mammogram, which is medically important. This data would stay on FC disk in the imaging center for 90 days, get replicated to FC disk in the hospital across town for 90 days, then the hospital copy would tier down to SATA for an additional 180 days and a copy would be made to LTO tape in the main datacenter across the country for long term archival. On the other hand, a negative mammogram, medically less important, would only remain at the imaging clinic for 14 days, but a copy would immediately get put on the LTO tape at the main datacenter for long term archival. It would have a different lifecycle than the positive mammogram.
- ▶ A second example is based on DB2 Content Manager. All files within the system with metadata tag = "ABC-ABetterCure trial" migrate from wherever they are to LTO tape, because the FDA is on our doorstep and wants copies. Alternatively, all files with metadata tag = "video" and metadata tag = "Father Christmas" migrate up to FC disk for sorting, because he just got sighted at The Magic Sledge Shoppe and we want some footage for the news.

The main point to understand is that applications accessing data with DB2 Content Manager must use the DB2 Content Manager interface.

Therefore, we can split ILM solutions by the way they classify and then manage files: solutions based on *file metadata*, and solutions based on *content metadata*. There is also a third category of solutions where the two kinds of metadata overlap: *hybrid system metadata*.

The second type of classification is the use that is made of the file system interface. A file system is an interface, as it allows access to underlying storage by translating named data objects, individual files in directories, to locations on physical media.

Data can remain visible in the file system even after it has moved, or migrated, to a different storage tier. Often this data is termed *near-line*. Higher level applications are not aware that the data has been removed from primary storage, as they keep on seeing it as if it were there. When applications refer to the data by opening the file, it is restored and accessed transparently; the application might notice a slight time delay if the file has to be recovered from tape. This function is often referred to as *transparent recall*.

Data can also be removed from the file system with an archiving application. The data is moved to the next level of the storage hierarchy and all references of it are removed from the primary storage. The application that uses the files must keep track of where the files have been stored and must initiate recovery when offline files are required.

There are various migration policies, indicated in the *migration policy* box of Figure 7-2 on page 162.

The first migration policy is *dynamic data classification and policy base migration*. This means that data gets classified automatically, based on rules such as size and age and, based on these rules, the data can be moved to another storage tier.

For installations looking for multiple levels of storage tiering, including disk and tape, automated migration and transparent recall, there are two solutions:

- ▶ For normal environments with up to millions of files per namespace, Tivoli Storage Manager for Space Management provides a solution. This product migrates data from the client to a Tivoli Storage Manager sever. When the data is accessed again by the application, it is recalled transparently.
- ▶ For very large environments with amounts of data in the order of tens or 100s of terabytes, you might require a solution such as HPSS, illustrated in “An ILM decision model” on page 160. This solution allows for access to large quantities of data in parallel from multiple nodes.

A second migration policy is *manual file selection*, when a system administrator defines a list of files to move to the next tier of storage. This can be implemented with a product such as Tivoli Storage Manager and the archive command. Files or lists of files can be archived in TSM and subsequently removed from the starting file system. The archive command also offers a grouping concept called a *package*: many files can be grouped together and a text description can be associated with the group. During retrieval, files can be searched by either file name or by the package description, making it easier to locate data in large sets.

Manual file selection can also be used with other products or operating system utilities to write to secondary storage devices such as a real or virtual tape device.

Query driven file selection is a hybrid approach. In our example we have TotalStorage Productivity Center for Data. This product can run commands on systems to create lists of files that match some arbitrary classification rule, such as age, owner, size or name. These lists can then be passed to an archival application to archive the files to the next tier of storage.

The next two solutions that we discuss are based on content metadata. The data might reside in a CIFS/NFS file system, or it might not have a file system interface to the data. The decision to move it to the next tier of storage is not based on file system attributes:

- ▶ One possible solution is the DB2 Content Manager family of products. This allows for policy driven data placement in the tiered storage hierarchy. It uses the standard DB2 Content Manager interfaces for data storage and retrieval.
- ▶ Another example solution, based on a CIFS/NFS file system interface, is the IBM Health Care and Life Sciences Grid Medical Archive Solution (GMAS). GMAS is an automated, self optimizing distributed grid storage solution. It allows a multi-campus hospitals to link disparate storage systems together and optimize utilization while offering full system redundancy and ensuring multiple copies of data are geographically separated. GMAS is Digital Imaging and Communications in Medicine (DICOM) content aware and allows for Hierarchical Storage Management/Information Lifecycle Management (HSM/ILM) based upon a file's metadata.

Data retention hardware decision model

Figure 7-3 illustrates a decision model to help you evaluate the optimal data retention hardware solution.

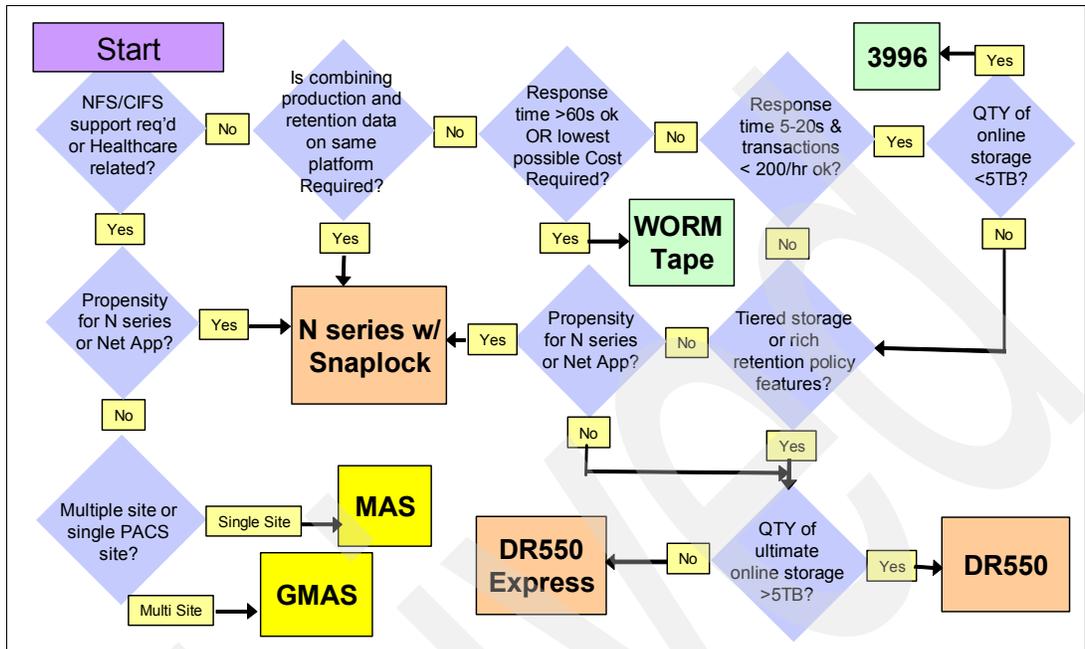


Figure 7-3 Data retention solutions hardware decision model

Note that this flowchart does not cover all scenarios and is provided for guidance only.

7.2 Best practices

In “ILM six best practices” on page 34 we discussed the six ILM best practices, or areas that installations are focusing on to address their ILM problems and requirements. In the following sections we illustrate each of these best practices in more detail.

7.2.1 Data rationalization

Data rationalization is used to establish valid groups of information in order to apply effective data management standards and policies. Conversely, after having established the valid groups, you are left with the invalid data groups, whose data you can eliminate or move and therefore reclaim and consolidate storage.

Data rationalization helps to address the following installation objectives and requirements:

- ▶ Controlling demand for storage
- ▶ Improving asset utilization
- ▶ Reducing hardware/software/storage personnel costs

To perform a data rationalization analysis, you must have tools and procedures that can help you understand your data, its age and use, and answer questions such as where is it and what data can be cleaned. You must be able to:

- ▶ Perform granular reporting that includes file system and database detail.
- ▶ Utilize data analysis techniques to determine where to reclaim and consolidate storage.
- ▶ Sustain improvements by connecting analysis output to process and policy improvements.
- ▶ Treat different data in different ways as necessary.

IBM Tivoli Productivity Center for Data (TPC for Data) is such a tool. TPC for Data allows you to gather and summarize space usage information and even perform historical trending analysis. Figure 7-4 illustrates this approach.

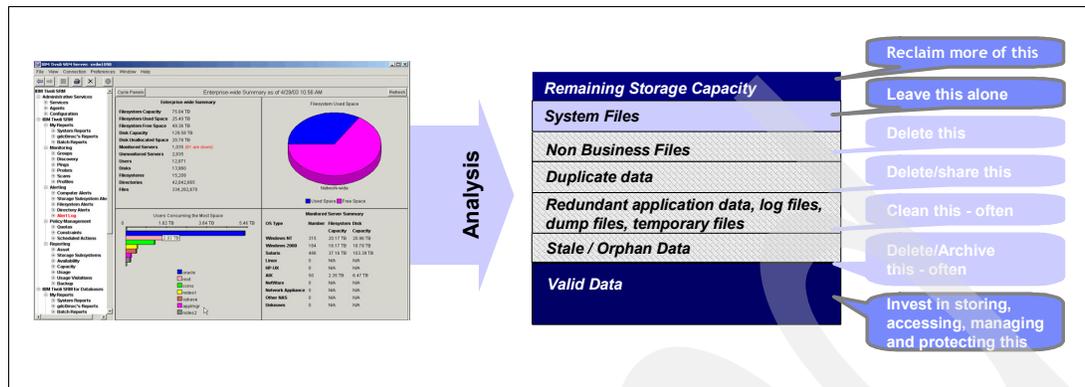


Figure 7-4 A data rationalization approach

We have been discussing the best practices that we have seen emerge from installations who were getting the best results out of their Information Lifecycle Management initiatives. Data rationalization is one of these best practices.

Data rationalization can be used for the following purposes:

- ▶ The first is to establish a valid groups of information so that we can apply effective data management standards and policies to each group.
- ▶ The second purpose for which data rationalization is used is to identify the invalid data groups, which might represent opportunities to reclaim and consolidate storage. The best practices that we suggest you use are related to exploiting the granular reporting capabilities, including file system detail. TPC for Data is an excellent solution to help with that. The foregoing figure shows examples of invalid data groups such as duplicate data, non-business files, or stale and orphan data (data that no longer has a valid owner assigned to it).
- ▶ The third purpose, from a best practices point of view, is to utilize the data analysis technique to determine where those opportunities are to reclaim and consolidate space, and quantify them.
- ▶ The fourth purpose would be to focus on sustaining improvements by connecting the output of the analysis to the process and policy improvements. The reason why there are invalid data groups in the first place is because policies are ineffective and processes are not as efficient as they should be.

For a more detailed discussion on how to use TPC for Data to analyze your storage environment, refer to the IBM Redbook: *ILM Library: Techniques with Tivoli Storage and IBM TotalStorage Products*, SG24-7030.

7.2.2 Storage virtualization

Storage virtualization simplifies storage infrastructure by combining the physical capacity from multiple disk and tape storage systems into a single logical storage pool that can be centrally managed. Storage virtualization can assist in ILM because virtualization allows you to move data transparently between storage tiers.

Storage virtualization helps address and achieve the following objectives:

- ▶ Infrastructure simplification
- ▶ Reduction of data migration efforts
- ▶ Improvement of asset utilization
- ▶ Reduction of hardware, software, storage, and personnel costs

These are some of the best practices that various installations are using to achieve the previously listed objectives:

- ▶ Deploy storage by storage pool, by creating and managing storage pools.
- ▶ Analyze environment to be virtualized to ensure support for heterogeneous environments.
- ▶ Use and leverage virtualization to enable transparent data migration.
- ▶ Automatically provision capacity from a single point of control including management of LUNs across enterprise, mid-range, and SATA technologies.
- ▶ Ensure that virtualization solution enables copy services from any storage array to any storage array.
- ▶ Automatically administer changes to the storage infrastructure through the use of a virtualized system management framework.

Figure 7-5 shows an example of hosts accessing a virtualized storage environment through a San Volume Controller device.

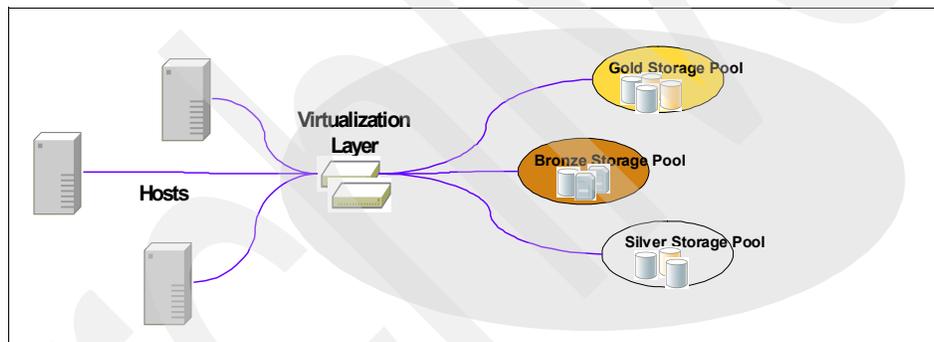


Figure 7-5 Virtualized storage infrastructure

This diagram shows different types of storage pools together with different service levels: gold storage, silver storage, and bronze storage. There is a virtualization layer implemented with a SAN volume controller. The SAN Volume Controller connects the storage to different hosts.

This gives the installation greater flexibility to increase asset utilization and also migrate data that is a part of any of these storage pools. If the installation changes the physical storage layers, this change does not disrupt anything that is happening on those hosts.

When we deploy virtualization, using our target architecture definition, we want to:

- ▶ Combine the capacity from different storage systems to a single storage pool.
- ▶ Make sure that we enable changes to the physical storage so that we have minimal or no impact to the applications running on the host.
- ▶ Reduce down time for planned and unplanned outages.
- ▶ Help increase storage capacity utilization and up time along with helping the administrator to be more productive and efficient.
- ▶ Help clients migrate data from the source to the target by leveraging virtualization.

7.2.3 Tiered storage

A tiered storage environment aligns variable cost hardware types with information classes and classes of service to create a variable cost storage environment. This definition might seem lofty but captures the essence of the problem: putting the data in the most appropriate place. The objectives are to:

- ▶ Maximize and sustain efficiency by improving current people, processes, and technologies being utilized to deliver storage services to the business.
- ▶ Define and implementing the appropriate storage strategy to address current and future business requirements.
- ▶ Make better use of existing information.

Some of the best practices that various installations are using to achieve the previously listed objectives are to:

- ▶ Align information with business requirements to accelerate movement of data off enterprise tiers to where it can be more cost efficiently stored and managed at the appropriate service level.
- ▶ Define variable cost technology types with corresponding information management policies.
- ▶ Establish well differentiated storage tiers and classes of service:
 - Leverage tape devices in the storage tiers.
 - Consider more than just hardware characteristics of storage.

Some aspects to consider are the use of tiered storage or separate storage levels to host different types of data, the classification and management information, the establishment of information management policies, and — last but very important — the development of a robust storage governance model to help sustain results. Figure 7-6 broadly outlines the steps required in classifying applications and their data and mapping them to the appropriate storage devices.

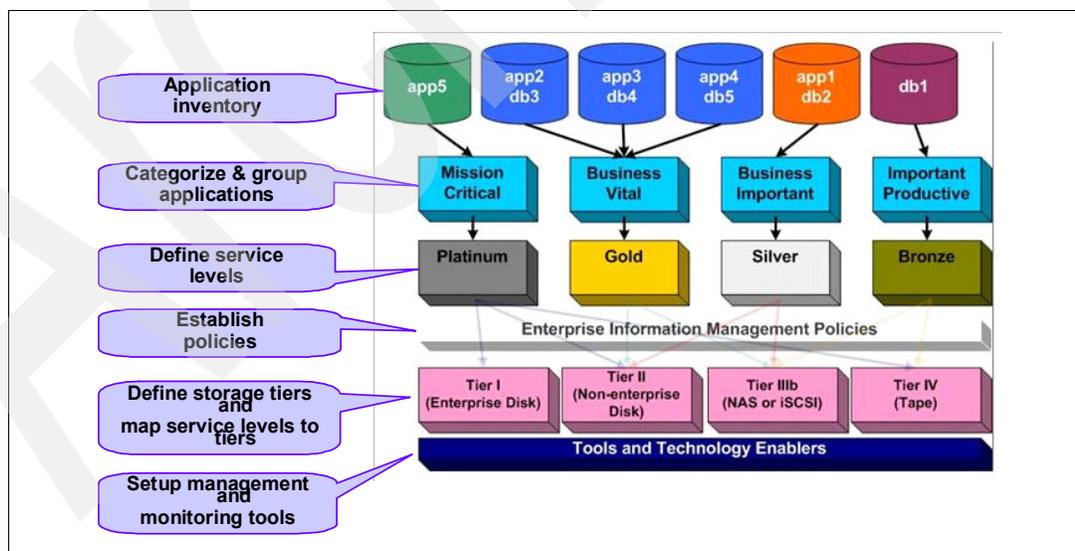


Figure 7-6 Tiered storage environment

We want improved information productivity so we can do that transformation of data and information to gain insight and make better decisions, as well as to ensure that we are aligning the variable cost hardware tiers with information classes and classes of service.

Here are some of the products and solutions that can assist in this area:

- ▶ TPC for disk is used to collect and analyze information, monitor the environment, and automate data movement between storage tiers based on pre-defined rules.
- ▶ System Storage disk devices such as DS8000, DS6000, DS4000 and N Series disk storage devices and SAN switches are used to build the tiered storage environment.
- ▶ System Storage tape devices and libraries such as IBM LTO Ultrium 3 and TS1120 with libraries such as 3584 are also important elements of tiered storage.
- ▶ Software products such as Tivoli Storage Manager server and the Space Manager component help to transparently migrate data between storage tiers.

7.2.4 Information management

Information management provides the ability to intelligently manage information. Here we consider information as the level above the pure file system data. Until now we have discussed the management of storage tiers and the inventory and proper management of individual files, where the entities were files in a file system. Information management is about managing data based on the information or content of the files themselves and making management decisions based on this content, because often the file name and other file system externals are not enough to make intelligent decisions for a specific data object.

Some installation objectives in this area, which are related to efficiency, are to:

- ▶ Maximize and sustain efficiency by improving the current people, processes, and technologies being utilized to deliver storage services to the business.
- ▶ Define and implement the appropriate storage strategy to address current and future business requirements.

Some of the best practices the industry is using today relate to these aspects:

- ▶ Categorize and classify information based on business attributes.
- ▶ Define information management policies: rules, procedures, and tasks for implementing the goals of the information infrastructure.
- ▶ Automatically apply information management policies to your information classes.
- ▶ Define and automate record retention policies and practices, disposition, and archival.
- ▶ Deploy intelligent search and discovery capabilities to ensure rapid and accurate retrieval when information is required.

Information management requirements are covered mostly by Enterprise Content Management products. Here is a definition of Enterprise Content Management:

A framework for creating, managing, integrating, Web enabling and delivering unstructured digital content across the enterprise and beyond - to employees, customers and trading partners – in a way that creates real business value.

Figure 7-7 defines and illustrates Information Management (IM), it is one of the *best practices* that clients focus on when they are trying to get the best results from their Information Lifecycle Management initiatives.

The differentiator between our approach and that of other companies is that we incorporate information management into our Information Lifecycle Management approach. Information lifecycle management typically focuses on managing cost and managing data growth. However, it is also about leveraging information assets to improve business decisions, reduce support cost, reduce the risk and cost of fraud, and streamline business processes by moving to more automated ones.

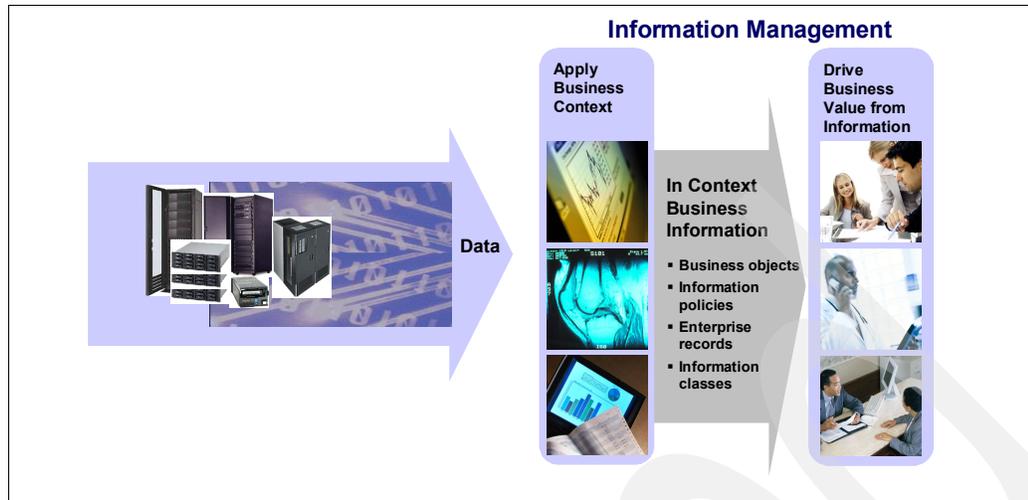


Figure 7-7 Information management definition

Therefore, information management is really all about integrating different kinds of data, then transforming the data into useful information so that it can be used to make better decisions more intelligently and more quickly.

Best practices include categorizing and classifying information based on business attributes, having policies in place, automatically applying those policies to the classes, and defining retention policies and practices, including disposing of data and archiving when appropriate. Then, the real lynch pin for us is deploying intelligent search and discovery capabilities to ensure rapid and accurate retrieval when information is required. Therefore, these important best practices are what constitute our key differentiators.

On the right-hand side of the foregoing diagram, we show applying the business context of information and allowing that to drive business value from the information, which can be used to make better and more intelligent decisions.

Traditional ILM is typically about managing costs and managing data growth. With Information Management (IM) software, it becomes:

- ▶ Not just managing costs, but truly leveraging your information assets to:
 - Improve business decisions by offering better access to accurate information.
 - Reduce customer support costs and improve employee productivity by helping users find and access information quickly.
 - Reduce the risk and cost of fraud through improved analytic capabilities.
 - Streamline business processes and costs by changing outdated paper processes to electronic ones.
- ▶ Not just managing the growth of data, but managing growing information to:
 - Integrate more data and content from different sources.
 - Transform this data into useful information and put it into context for use by analytical applications.
 - Analyze this transformed information in real time to enable intelligent decisions quickly

IM lets you gain more value from information investments because it allows you to create business value from existing information assets.

7.2.5 Storage governance model

An ILM governance model is comprised of process, organization, technology, service management, and governance components. The objective of the governance model is to sustain the value of the initial ILM implementation in time, to govern the storage environment so that it continues to follow the ILM rules that were laid down.

Here are some of the best practices that installations are currently leveraging in this area:

- ▶ Assessing which opportunities will have the greatest impact: process, organization, service management or technology.
- ▶ Deploying a Storage Governance Model to improve process, organization and technology through standardizing activities, roles, responsibilities, inputs, outputs and controls.
- ▶ Enhance and automate information management and storage specific processes to sustain improvements.

Figure 7-8 shows the inner relationship of those different pieces of organization, process, and technology, with storage service management being in the middle and governance connecting everything together. From a best practices point of view, we focus on assessing which opportunities will have the greatest impact, process, organization, or technology. Different approaches will have different results in different installations.

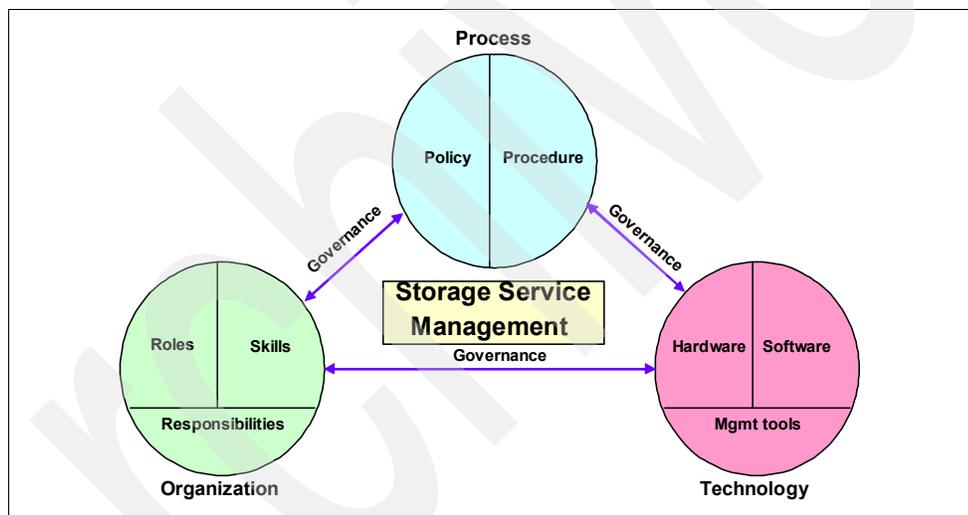


Figure 7-8 The governance relationship model

The first major aspect here is that, most often, installations tend to focus on the technology components and not the other ones. In such a case, they cannot sustain the results.

The second aspect is to deploy the governance model to help improve process, organization and technology by standardizing activities, roles, responsibilities, inputs, outputs, and controls. That is a very important aspect to making this work.

The third major aspect of best practices with respect to a governance model is to enhance and automate information management and storage specific processes in order to sustain the improvements. It is often hard to sustain the results that you might gain out of making technology improvements. We see this as a common pitfall that installations go through when they are not focused on the sustaining aspects.

Figure 7-9 shows a structured approach to building an ILM solution.

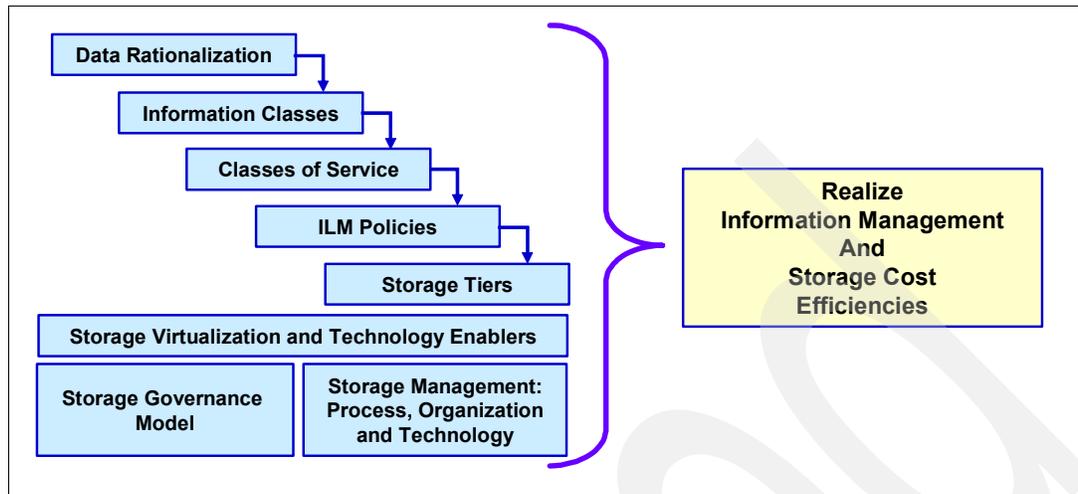


Figure 7-9 ILM structured approach

When we look at the best practices deployed, they are actually aligning with virtualization, tiered storage, the process organization technology and governance model, as well as information management.

Here is our suggested approach: You start by defining information classes and classes of service. Next, you establish information policies. Then, you design storage hardware and infrastructure software tiers. After that, you design and implement storage virtualization. Finally, you can improve storage management inefficiencies and establish the governance model to enable sustaining of the benefits.

This picture tends to connect the dots between the various things installations can do to realize information management and storage cost efficiencies.

What we find is that in order to get storage tiers, installations tend to look at putting policies in place and defining segmented service levels — basically, a portfolio of service offerings to offer their different application owners, to make sure they are able to offer mission critical applications the best level of service. The next level of applications get very good service, but not as much as the most critical application, and so on, right down the line.

So many installations have looked at tying together classes of service and policies in conjunction with storage tiers in order to gain and maximize those efficiencies. Information classes refers to one of the techniques used to help figure out how the segment of your application and data works — the validated group of data, by the way, so that you can think about the service levels, the policies, and the tiers that are required to support them.

Also shown in this picture is the data rationalization step, where installations are focused on initially on cleaning up their data to separate out the invalid data from the valid data, so that only the valid data goes into the information classification process.

When we examine activities that we see a lot of installations implementing, this seems to involve classifying information, classes of service, policies, tiers, and virtualization. Installations do this in conjunction with looking at the governance model and storage management.

Other installations prefer to choose a lower entry point, which is data rationalization. Therefore, they might want to focus on just the cleanup of the invalid data.

7.2.6 Archiving and information retention

Archiving and information retention enhances systems performance while enabling installations to better manage risk and streamline regulatory compliance. Some installation requirements in this area are to:

- ▶ Improve efficiency:
 - Enhance systems
 - Enhance e-mail performance
- ▶ Manage risk and streamline compliance:
 - Reduce organizational risk
 - Comply with governmental regulations

Here are some of the common best practices in this area:

- ▶ Use non-erasable, non-rewriteable technologies to help protect against accidental or intentional alteration and/or deletion of information prior to its expiration.
- ▶ If you have large volumes of data, greater than tens of TB, you should utilize tiered storage in archiving and retention strategies to significantly reduce TCO.
- ▶ Evaluate performance and scalability prior to deployment.
- ▶ Utilize event-based retention management for data that has unspecified retention periods. Examples of these kinds of data include employee and customer records, patient records, and insurance policies.

Archiving and information retention basically enhance system performance, but also enable an organization to better manage risk and streamline regulatory compliance. From a best practices point of view, this means employing non-erasable and non-rewriteable technologies to protect against either accidental or maliciously intentional alteration or deletion of information before it is supposed to be deleted.

Installations who have large volumes of data, greater than 10 TB, should think about using a tiered storage in archiving and retention to significantly reduce total cost of ownership and improve the efficiency of the environment.

It is important to evaluate performance and scalability prior to deployment to make sure that you are going to get the desired results before you start deploying anything.

Finally, we suggest that you utilize event based retention management for data that has unspecified retention periods. For instance, you can look at some event that is going to trigger being able to archive or retain data. An example might be employee and customer records, patient records, or insurance policies or insurance claims.

Figure 7-10 shows a variety of different data types and different mechanisms that we use. The goal is to move to the middle of the diagram, where we can create, store, search, distribute, manage, and retain data in order to leverage the complete solution. This diagram also shows some of the hardware components that will make this possible.

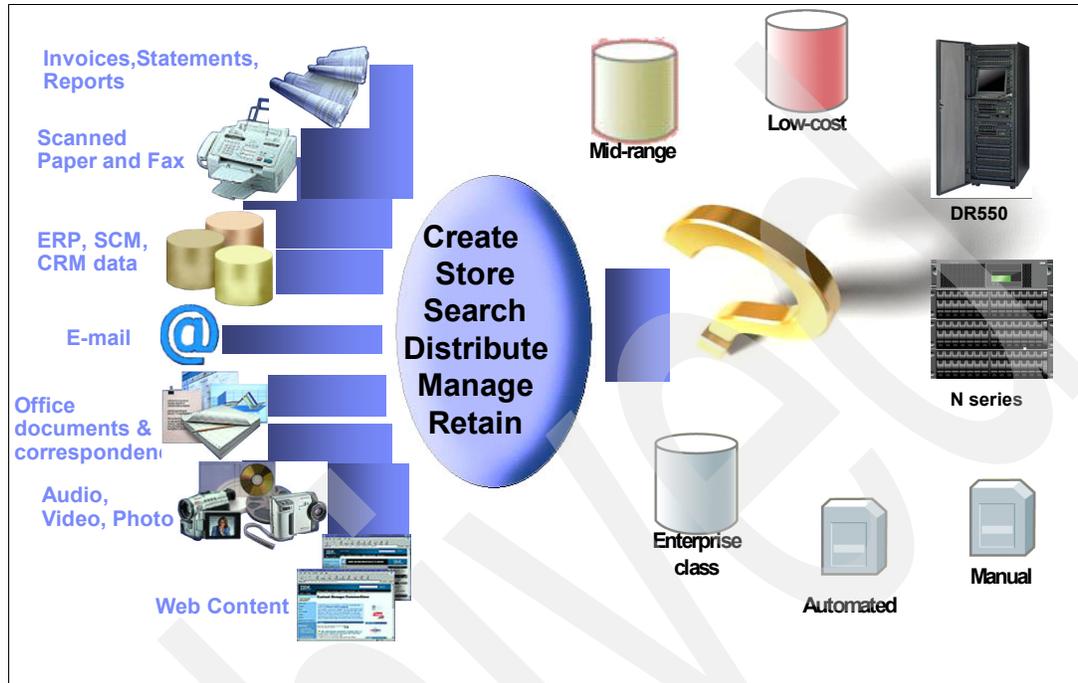


Figure 7-10 Where to archive content?

There are specific software archiving solutions for each type of application. For example, for SAP archiving we can focus on archiving data and gain some benefits by reducing the growth of the database, improving the backup and restore and reorganization capabilities, and also making sure the performance improves by getting rid of the clutter of some of the old data.

We can also do the same thing with documents and archive those documents, incoming and outgoing. We can leverage the SAP document finder when we do that archiving, and also realize that this supports My SAP solutions on all the enabled platforms and databases that SAP works on.

If we consider all the components that are used, the solution components from a hardware, software point of view are the DR550, Content Manager, and also potentially Common Store for SAP, and this solution is certified by SAP.

Another solution area is e-mail archiving. With e-mail archiving, which is growing in popularity with many installations, we offer an integrated e-mail archiving solution to help clients do a number of things.

We help store, retrieve, manage, discover, retain, and dispose of the e-mails as appropriate to support compliance and risk, and offer a records management capability to automate declaration and classification of e-mail or attachments as corporate records. Flexible automated archiving policies can be based on a number of different characteristics. You should make sure that you have storage management that supports the ability to not erase or not rewrite, to support long term retention requirements. Also, make sure that you can archive either complete mail documents or just the file attachments.

The components of our solution, shown on the right side of the diagram in Figure 7-10, are:

- ▶ DR550
- ▶ Content Manager
- ▶ Common Store for Microsoft Exchange and Lotus Domino

Mail retention solutions address some commonly known facts:

- ▶ 90% of all e-mails have no attachments.
- ▶ 10% of all e-mails have attachments:
 - They occupy about 75% of the entire mail box.
 - They are created by applications, including graphics.
 - They grow with every application release.
- ▶ Old e-mails have the following characteristics:
 - They experience little access.
 - They are kept for possible later look-up.
 - Deletion is no option for the user.

Princeton Softech's Optim Solutions allow companies to automate the process of storing archived data according to retention policies and the data's current stage in its lifecycle. With HSM you can use various storage tiers to store the older data. You can select less expensive, secondary storage media to store Archive Files and reuse expensive primary disks for current application data. Together, HSM and Princeton Softech's Active Archive Solutions enable companies to automate the migration policies set by administrators, according to a company's ILM strategy.

For non-erasable and non-rewriteable storage devices, you can consider using either the IBM DR550 or N series storage with the SnapLock feature. The decision depends on two key aspects:

- ▶ Does the application you plan to use support one of these storage devices?
- ▶ Do you require only disk storage or multiple disk and tape tiers? In the latter case, you should use the DR550.

Here is a schematic way of classifying your data's archival and retention requirements:

- ▶ Retention cycle: Does the information have to be retained for a specific period for a corporate governance or regulatory purpose?
- ▶ Disposition cycle: After the retention cycle is complete, should the information be disposed of completely archived to a lower-cost media?
- ▶ Archival cycle: Does the information have to be archived for long periods? If so, does this archival require to be stored separately from the original?
- ▶ Access frequency: How frequently or infrequently is the information accessed after it is created? Will it be write once read many, or write once read rarely, or will it have a more active access frequency?
- ▶ Read/write performance cycle: Based on the access frequency of the data, what is the required performance for both read and write operations? What technologies are appropriate for these requirements?
- ▶ Read/write permissions: Does the information have to be stored on non-erasable, non-rewriteable media?
- ▶ Recovery performance cycle: How quickly does the information have to be recovered?
- ▶ Security issues: How will the compromise of this information at different points in its lifecycle affect the business?

Answering these questions will help you choose the appropriate infrastructure for data archival and retention.

7.3 The IBM approach with SMCD-ILM

IGS offers a comprehensive set of services to assist installations define and deploy and maintain their ILM strategy. Here, we discuss one in particular: IBM Systems Management Consulting and Design Services - Information Lifecycle Management (SMCD-ILM).

In four structured, flexible steps, SMCD-ILM can help your organization align the business value of its information with the most appropriate and cost-effective IT infrastructure, from the time information is conceived through its final disposition. Some of the challenges that SMCD-ILM can assist you with are:

- ▶ Data growth management: SMCD-ILM helps reduce redundancies and inefficiencies by evaluating and designing storage strategies that reflect data's business value over time.
- ▶ Cost control: SMCD-ILM helps you to accurately compare data's value against the cost of a proposed storage solution, and choose the most cost-efficient option.
- ▶ Compliance: SMCD-ILM helps you to develop an Information Lifecycle Management program designed to assist with automating compliance with your industry's document conservation requirements.
- ▶ Risk management: SMCD-ILM is designed to help you accurately assess your data's business value over time, so that you can protect data and manage risk of data loss more efficiently.
- ▶ Migration path: SMCD-ILM recommendations can be implemented with confidence, because they start with your existing infrastructure, incorporate open system principles and are ITIL compliant.

SMCD-ILM tells you what you have to do, how you must do it, and then helps you create a plan for getting it done. Questionnaires, interviews and workshops incorporate your input every step of the way. The resulting report maps your organization's different data types to appropriate storage technologies.

Then it identifies what you must have to help translate that map into customized strategies for success: policies for data value analysis; processes for automated, transparent lifecycle management; practices for security measures, risk reduction and compliance; and tools for evaluating future technology investments.

What you and your organization come away with is a customized blueprint for Information Lifecycle Management storage infrastructure that can be more effectively and efficiently implemented, and that is designed to leverage your existing investment in storage, remove redundancies and make data more accessible to the people who require it.

The customized ILM design that we develop for you comes together in four steps, as shown in Figure 7-11.

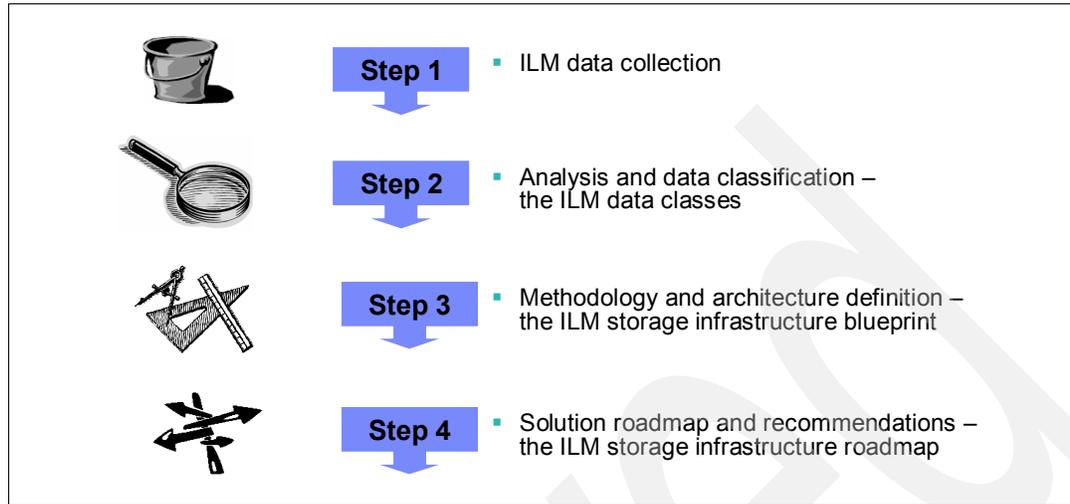


Figure 7-11 The SMCD-ILM four step approach

Step 1. ILM data collection: We work with your IT staff to collect the information and data related to the environment in the scope of the analysis.

Step 2. Analysis classification — the ILM data classes. We define the ILM classes of data, the ILM policies for each class, and the requirements for how data in each class should be stored throughout its lifecycle. At this step, we also identify opportunities for quick wins, such as data cleanup, rationalized space usage, and adaptive capacity plans.

Step 3. Methodology and architecture definition — the ILM storage infrastructure blueprint. This design stage defines the storage technology, storage management processes and organization required to support the data classes and the ILM policies established in Step 2. The resulting storage architecture is vendor-neutral.

Step 4. Solution roadmap and recommendations — the ILM storage infrastructure roadmap. The final step provides an action plan for implementing ILM storage infrastructure. We start by mapping known vendor ILM solutions to your defined architecture and selecting the “best fit” solution. We then identify gaps between your current and target environments and create a comprehensive deployment program for change.

SMCD-ILM is comprehensive, but still extremely flexible. Its four steps can be implemented as a continuous program, or as necessary, and can provide you with the data infrastructure management options you must have, when you require them.

Table 7-1 shows some of the possible data infrastructure management study options.

Table 7-1 Data infrastructure management study options

Your situation:	The SMCD-ILM formula to match:
I have a number of storage issues but do not know where to start.	A framing workshop (part of step 1) to review your current environment, make high level suggestions on areas to focus, frame an approach to address storage/ILM issues.
I know what the problem is. I want to solve it, and identify some quick wins and a further course of action.	An assessment (steps 1 and 2) identifying the opportunity for quick wins and long-term benefits.
I must have a design and a transition plan to ensure ROI of investments in technology and minimize the risks.	A solution design and transition plan (steps 1 through 4) that is designed to satisfy your organization's wants and requirements related to storage infrastructure.



IBM Tivoli Storage Manager best practices

In this chapter we discuss Tivoli Storage Manager, SSAM, and DR550 infrastructure sizing as well as protection of the data stored in Tivoli Storage Manager, when this data is not a backup, but probably the last valid copy of the data.

We cover the following topics:

- ▶ Determining requirements and sizing the Tivoli Storage Manager environment
- ▶ Protecting against local media failure and disaster
- ▶ The life of an object stored in Tivoli Storage Manager and SSAM

8.1 Sizing the Tivoli Storage Manager environment

This section discusses the sizing of a Tivoli Storage Manager data retention solution. How and where do you start sizing such a solution? You must determine the requirements as input information, and from this you will be able to estimate the Tivoli Storage Manager server environment that can satisfy your requirements.

Based on this information, we discuss how to size the Tivoli Storage Manager environment with information such as:

- ▶ The amount of storage required for the data
- ▶ The size of the Tivoli Storage Manager database
- ▶ What storage media should be used to satisfy the recovery requirements

8.1.1 Determining business requirements

You must understand the business requirements for sizing such a solution, and based on these requirements you can start sizing the Tivoli Storage Manager environment. Basic questions for which you must provide an answer are as follows:

- ▶ How much data must you store? For how long?
- ▶ Do you require point-in-time copies?
- ▶ Are there specific storage requirements such as use of WORM devices?
- ▶ What are the recovery requirements in terms of speed and frequency of access?
- ▶ Do you require off site disaster recovery capabilities?

One fundamental aspect is determining the total amount of data to be stored in the Tivoli Storage Manager environment. Often this quantity is not known with precision, and only informed guesses and estimates are available. IBM Tivoli Productivity Center for Data (TPC for Data) can help with the estimation of data by profiling all of your clients, databases, file servers, and NAS devices.

You should start by evaluating the application or applications that require to store their data into the Tivoli Storage Manager server environment and determine the amount of changes each day and criticality of the data. Also try to determine how the application will access the data. Example questions you can ask are:

- ▶ What is the total amount of data to be backed up if you are backing up for the first time?
- ▶ What is the average file size?
- ▶ What portion of the total amount of data is database data?
- ▶ What is the average file or object size?
- ▶ Is the data to be compressed by the Tivoli Storage Manager client application before being sent to the Tivoli Storage Manager server?

Compression of data by the client prior to sending it to the server will reduce the total amount of storage space required by the compression factor.

- ▶ Is the data compressed at the storage device?
- ▶ How often will a full backup of the database data be done?
- ▶ How often will the database logs be backed up?
- ▶ What is the predicted change rate for the file data?

- ▶ Are there specific access time requirements, such as time to restore a given object?

The Tivoli Storage Manager server supports multiple tiers of storage, as discussed in 4.1.2, “Tivoli Storage Manager storage management” on page 82. Data that is frequently accessed or requires fast access should reside on disk. Data that can tolerate longer access times and is accessed infrequently can be located on tape devices that have a lower cost than disk devices.

- ▶ Are WORM storage devices required?

Determine business specific requirements for non erasable, non rewriteable storage devices. These WORM storage requirements depend on your interpretation of rules and regulations.

- ▶ Should the data be protected for disaster recovery?

- ▶ Should the disaster recovery copy data be placed on WORM devices?

Data that is archived on to storage devices managed by Tivoli Storage Manager is often the last valid copy of the specific data object. Therefore, protection against unforeseen events from individual storage media failures to full scale disasters should be evaluated. If the data is to be protected for disaster recovery does it require the same level of storage media protection, for example WORM, as the primary data?

- ▶ Are there specific disposition requirements?

Disposition controls the fate of data after it has expired from the Tivoli Storage Manager server. Are there requirements that the actual data should be physically deleted from storage or is it sufficient for the Tivoli Storage Manager metadata to expire.

- ▶ How will your retention requirements evolve in the following years?

The last is possibly the hardest question to answer but is probably the most important. Data lifecycle management and data retention often imply that the data be kept for periods of years, possibly 5 to 10 years. On such a time scale the application or applications requirements will change: the data from an individual application will probably grow, retention requirements will probably change, and new applications will probably require Tivoli Storage Manager services.

This means that the infrastructure that you will build initially will probably require reviewing in the future.

A second, very important, aspect is that the retention requirements for the data can and will probably exceed the lifetime of the underlying hardware and software solution. Therefore, there are two key aspects to consider. The first is the ease of migrating your Tivoli Storage Manager infrastructure to new hardware, either server or storage devices, without interrupting service or losing archived data. The second relates to the future availability of the software middleware: Tivoli Storage Manager (formerly known as ADSM) was introduced in 1993, and as of today has been on the market for 13 years. IBM over the years has constantly evolved, improved and maintained Tivoli Storage Manager and plans to continue to do so in the foreseeable future.

8.1.2 Sizing the Tivoli Storage Manager environment and selecting media

After you have determined your business requirements and translated these requirements into technical requirements, you can start with the Tivoli Storage Manager environment sizing exercise and choice of media. The aim is to determine such characteristics as:

- ▶ Size of Tivoli Storage Manager database
- ▶ Size of Tivoli Storage Manager storage pools
- ▶ Number and type of storage pools and capacity of each storage pool

- ▶ Types storage devices for the Tivoli Storage Manager storage pools
- ▶ Number of slots and drives in external tape libraries
- ▶ Type and capacity of the server based on the amount of data transferred to the server each day:
 - Amount stored for archive
 - Amount retrieved by the application

Sizing of the Tivoli Storage Manager database and log

The database and recovery log should be created at installation time because they are required by Tivoli Storage Manager to start. Tivoli Storage Manager offers advanced functions to manage its database. By using these functions, you can perform the following actions on both the database and the recovery log without impacting Tivoli Storage Manager server availability:

- ▶ Expand the size, or reduce it if desired
- ▶ Move parts or all of the database or log to different disk storage devices
- ▶ Mirror or remove mirror copies on disk

These advanced functions allow the flexibility to do a rough sizing of the Tivoli Storage Manager database and log and change total capacity and underlying storage devices as required, without service interruption.

The size of the Tivoli Storage Manager database depends on the number of files that are stored in it, and the method by which the server manages them. Each entry represents one individual object in the Tivoli Storage Manager storage hierarchy. If you can estimate the maximum number of files that might be in server storage at any time, you can estimate the database size from the following information:

- ▶ Each stored version of a file requires about 400 to 600 bytes of database space.
- ▶ Each cached or copy storage pool file requires about 100 to 200 bytes of database space.
- ▶ Overhead could require up to 25% in additional space.

Cached copies are used for data that has migrated to the next storage pool in the Tivoli Storage Manager hierarchy but is still available for read access in the original storage pool. If a request comes in for a cached file, it is satisfied from the original storage pool. If space is required in the storage pool, the cached files are invalidated and the space can be used by new files.

In the example given later, the computations are probable maximums. In addition, the numbers are not based on the use of file aggregation. In general, aggregation of small files reduces the required database space.

Assume that we have determined the application requirements shown in Figure 8-1, we have three applications, and for each we have determined the inputs: the average number of objects stored each day, the average size of the objects, and the retention period of the objects. Note that in the example we are not considering event based retention, for simplicity of exposure. From such data we can easily calculate the following values:

- ▶ Server archive GB day represents the amount of GB stored on the server each day. It is calculated as the average number of objects per day multiplied by the average object size.
- ▶ Total storage pool GB represents the total storage pool capacity at the end of the initial startup or ramp up period. It is calculated as the average number of objects per day multiplied by the average object size multiplied by the number of days the objects have to be retained. Different applications can have different retention periods, therefore, the value is calculated application by application.

Note: The initial startup or ramp up period is defined as the period it takes the Tivoli Storage Manager server to reach the steady state, and it corresponds to the retention period for the data. An application with a retention of three years will reach the steady state after three years. At the end of year one, 33% of the data is stored in Tivoli Storage Manager; at the end of year two, 66%; and at year three, 100%. After year four, we still have 100% because year one data will have expired.

- ▶ Storage pool GB per year represents the amount of storage that must be added each year to satisfy data retention requests. If the application has a ramp up period of 5 years, only 20% of the cumulative storage must be deployed in the first year.
- ▶ Database GB per year represents the yearly growth in size for the Tivoli Storage Manager database.
- ▶ Database GB per year +25% represents the yearly growth in size for the Tivoli Storage Manager database, factoring in the recommended 25% overhead.
- ▶ Total database size GB represents the total Tivoli Storage Manager database size after the ramp up period.
- ▶ Total database size GB + 25% represents the total Tivoli Storage Manager database size after the ramp up period, factoring in the recommended 25% overhead.

Table 8-1 Tivoli Storage Manager database sizing requirements and results

		App1	App2	App3	Totals
INPUTS	Average objects per day	10,000	25,000	8,000	N/A
	Average object size	1	.5	.2	n/a
	Retention on disk years	3	2	3	n/a
	Retention years	5	7	10	n/a
	Total objects	18,250,000	63,875,000	29,200,000	111,325,000
	Average database entry in bytes	600	600	600	n/a
RESULTS	Server archive GB per day	10.0	12.5	1.6	24.1
	Total storage pool GB	18250	31938	5840	56028
	Storage pool GB per year	3650	4563	584	8797
	Database GB per year	2.2	6.6	1.8	9.4
	Database GB per year + 25%	2.7	6.8	2.2	11.8
	Total database size in GB	11	38.3	17.5	66.8

These initial sizing calculations do not include the additional database space required for caching the objects, or a recommended additional space of 25% for database overhead. Most importantly, these calculations do not take into account new, unplanned workloads that might be added to the Tivoli Storage Manager environment.

Important: Notice that in our example we have not discussed Tivoli Storage Manager client compression, performed on the client before sending the data to the server. For our sizing exercise, we assume that average object size refers to already compressed data.

In the example we discussed previously, we can see that the Tivoli Storage Manager database requires 66.8 GB of storage space, we add 25% overhead to this value, and we have around 83 GB of total Tivoli Storage Manager database space required.

Figure 8-1 shows the Tivoli Storage Manager database growth over a period of time. We can clearly see the ramp up period for our database: Based on our assumptions, the database will reach the target size of 83 GB only on the 10th year; at the end of the 4th year we still require less than 50 GB of database space. Using this kind of information, you might decide to provision storage resources for the Tivoli Storage Manager database gradually, as they are required.

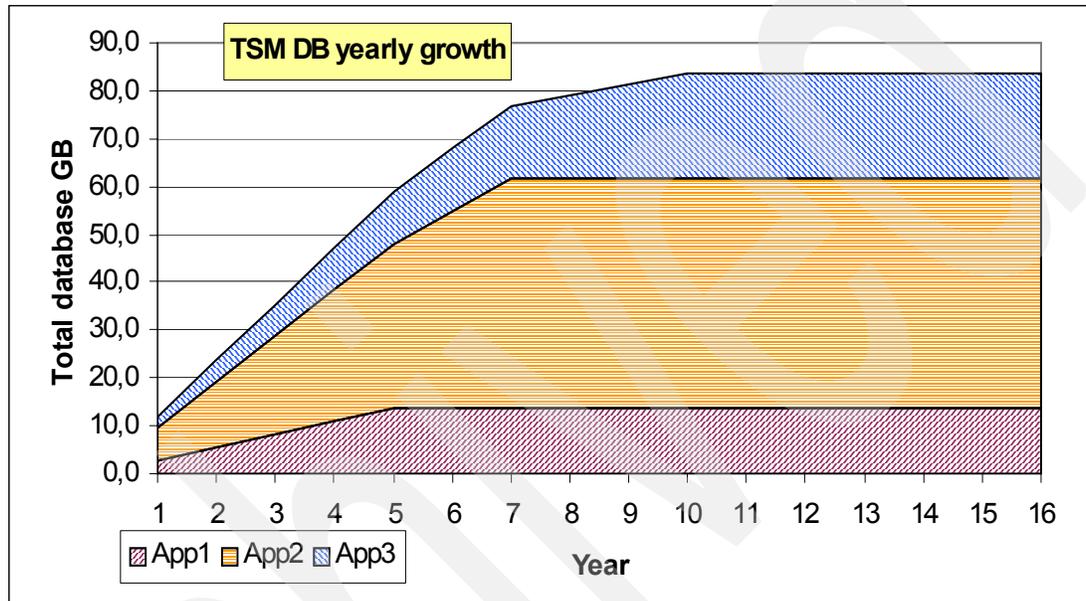


Figure 8-1 Tivoli Storage Manager database growth over time

To size the Tivoli Storage Manager database log, we have to understand Tivoli Storage Manager transactions: a transaction is the unit of work exchanged between the client and server. The client program can transfer more than one file or directory between the client and server before it commits the data to server storage. Therefore, a transaction can contain more than one file or directory. This is called a transaction group.

Tivoli Storage Manager provides a TXNGROUPMAX server option that allows you to specify an upper limit to the number of files or directories contained within a transaction group. It is possible to affect the performance of client backup, archive, restore, and retrieve operations by using a larger value for the TXNGROUPMAX option. You can use the TXNGROUPMAX option to increase performance when Tivoli Storage Manager writes to tape. This performance can be considerable when a user transfers multiple small files. If you increase the value of TXNGROUPMAX by a large amount, you should monitor the effects on the recovery log. A larger value can increase utilization of the recovery log, as well as an increased length of time for a transaction to commit.

The number of transactions affect how large you should make your recovery log. As you add more clients and increase concurrent transactions, you can extend the size of the log. The Tivoli Storage Manager database can be configured in roll-forward mode. In this mode, Tivoli Storage Manager performs transaction logging; all completed transactions are saved in the recovery log and these use up space. The log is then saved periodically; this saved copy is known as an incremental database backup. Contrast this with Tivoli Storage Manager running in normal mode, where transactions are recorded in the recovery log until they are committed, and then the recovery log space is reused.

The advantage of running in roll-forward mode is that the Tivoli Storage Manager database can be recovered from a full database backup, done on a periodic basis, and then the incremental database backups can be reapplied, followed by all committed transactions in the Tivoli Storage Manager recovery log, if available. This allows for the database to be recovered to the most current state and not to the last point in time full backup. In roll-forward mode you should consider how often you perform database backups. In this mode, the recovery log keeps all transactions since the last database backup and typically requires much more space than normal mode does.

Note: By default, the DR550 database is configured in roll-forward mode.

To determine the size that the recovery log should be in roll-forward mode, you must know how much recovery log space is used between database backups. For example, if you perform daily incremental backups, check your daily usage over a period of time. A suggested starting size setting for the recovery log in roll-forward mode is around 5 GB.

For additional information on sizing the Tivoli Storage Manager recovery log, refer to the chapter, “Managing the Database and Recovery Log”, in the *Tivoli Storage Manager Server Administration Guide* for your operating system platform, which can be found at:

<http://publib.boulder.ibm.com/infocenter/tivihelp/v1r1/index.jsp>

Determining the size and media for the storage pools

After determining the size of the Tivoli Storage Manager database, you must determine the number and type of Tivoli Storage Manager storage pools. For this, we discuss:

- ▶ Sizing the storage pools
- ▶ Designing a storage hierarchy
- ▶ Choosing storage media for the storage hierarchy
- ▶ Tape considerations

Refer back to the model illustrated in Figure 8-1 on page 183. To determine storage pool capacity, we have to know how much data we will receive: number of files and average file size, and also the retention time for that category of data.

Given these inputs, we can easily calculate the total storage pool size for each set of data; our model shows three sets of data: app1, app2, and app3. For each one, we calculate the total storage pool size as the number of objects stored each day multiplied by the average object size, multiplied again by the number of days the objects must be retained. Figure 8-2 shows the amount of storage required for each storage pool and how the cumulative amount of storage grows during the ramp up period.

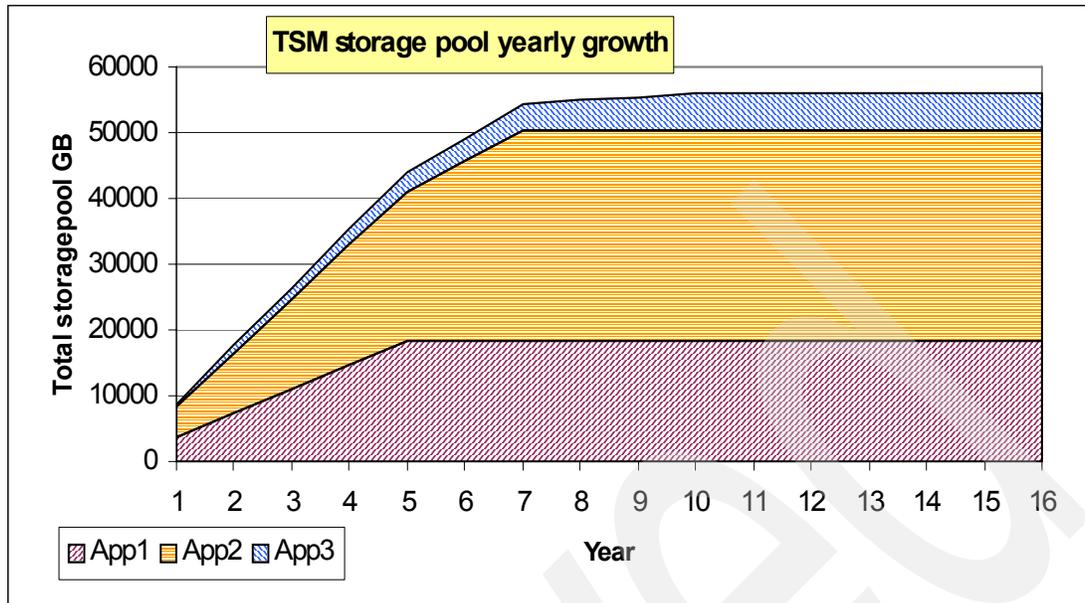


Figure 8-2 Tivoli Storage Manager storage pool yearly growth

From Table 8-1 on page 183, we see that the app1, app2, and app3 storage pools require 18, 32, and 6 TB at the end of the ramp up period. You now have to design a storage hierarchy to store this data. Data should initially be stored on disk storage devices and transition to a less expensive media such as tape after a certain period. In the following discussion we consider a two tier storage hierarchy with disk and tape. In our example, app2 requires 32 TB of storage pool storage with around 4.5 TB being stored each year.

The decision you must make is how long to keep the data on disk before migrating it to tape. The advantage of keeping data for a shorter on disk is that you require less potentially expensive disk storage. Therefore, why not send all data to tape as soon as possible? This depends on the amount of archived data that is then accessed by the application: how often is the data accessed, and how long after the date it was archived? When data is stored on tape, access is not immediate as it is on disk. Because you require tape mount and positioning, you can realistically expect a one to two minute access time.

We recommend that you store frequently accessed data on disk and less frequently accessed data on tape. In our example, shown in Figure 8-2, we again have our three applications and, based on business requirements, we have defined a retention on disk value for each individual application.

Table 8-2 Application retention requirements

		App1	App2	App3
Inputs	Average objects per day	10,000	25,000	8,000
	Average object size in MB	1	.5	.2
	Retention on disk in years	3	2	3
	Retention years	5	7	10

Based on these inputs, we estimate the amount of disk and tape storage space required for each application and also the cumulative disk and tape space, as shown in Figure 8-3.

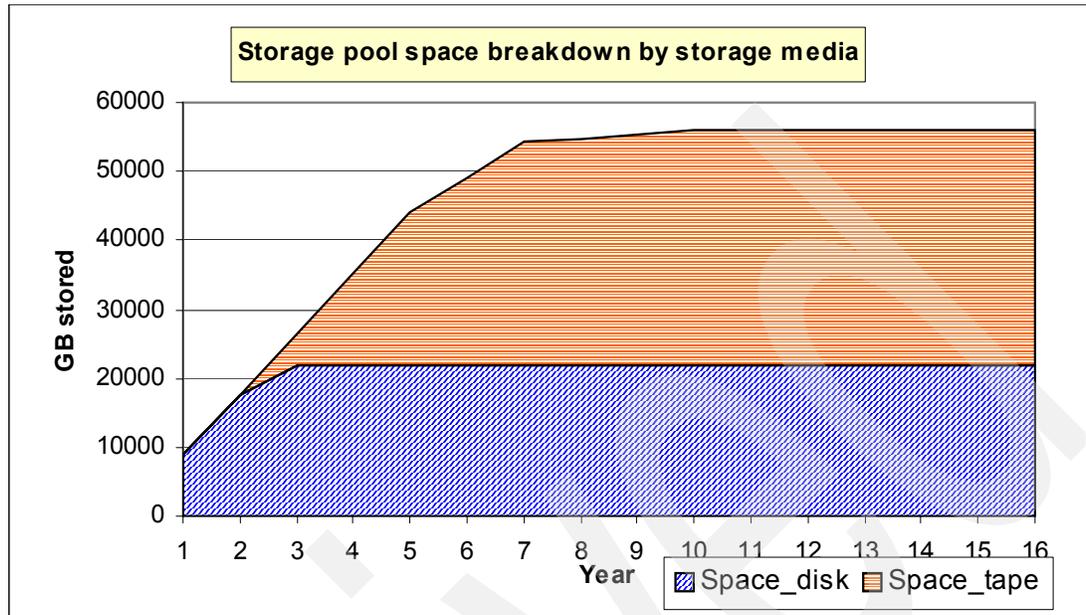


Figure 8-3 Storage pool space breakdown by pool

When reading this chart, you can determine various interesting aspects:

- ▶ The total storage pool space required, based on our retention rules, is just under 60 TB.
- ▶ You require around 20 TB of disk storage pool space.
- ▶ At the end of year 1 you use 10 TB of disk, and at the end year 2 you use 20 TB.
- ▶ Tapes, for Tivoli Storage Manager primary storage pools, will start to be used sometime in year 3.
- ▶ Use of tapes will grow constantly until year 7, when it falls off as old data expires.

Therefore, you have determined the total disk and tape capacity required. The number of tape cartridges for the primary pool can easily be calculated by dividing the total storage pool capacity requirement by the cartridge capacity. Adding a reasonable contingency, we suggest a value of around 20% of total cartridges.

The number of tape drives is more difficult to determine. A minimum value to allow for Tivoli Storage Manager efficient operation and functionality is two drives, which allow you to:

- ▶ Process two restore requests in parallel.
- ▶ Create multiple copies of data for both onsite and offsite use.
- ▶ Perform tape reclamation and consolidation operations.
- ▶ Ensure availability in case of the failure of one drive.

Note: We highly recommend that you use at least three drives for any Tivoli Storage Manager configuration to allow for tape hardware failures and maintenance.

Often data retrieval processes are initiated by a person who requires access to one or more objects stored into Tivoli Storage Manager, probably using some kind of content management application. There might be multiple parallel requests, and when the number of requests is greater than the number of available tape drives, the requests are serialized: the first requests access the available drives, and the remaining requests are put into a queue on a first come, first served basis.

A person accessing the data on a tape has to wait for tape mount and positioning. This can lower their productivity because of the time spent waiting for an available tape drive and then the data.

The frequency of data access often decreases with age, therefore, the older the data gets, the less frequently it is accessed. The idea is to store frequently accessed data on disk and less active data on tape, thus minimizing the probability of accessing data on tape and consequently guarantee reasonable average response time Service Level Agreements (SLAs).

A detailed sizing model of the number of drives for a given number of requesters (users) and servers (tape drives) is outside of the scope of this book. It requires the use of queue theory.

As a guideline for the number of drives to satisfy data retrievals, we suggest that you use the greater of these two values, independently from the data rate of the drives:

- ▶ A minimum of two drives — however, a minimum of three is highly recommended.
- ▶ The peak number of requests per hour that require a tape mount divided by 30, on the assumption that the drive and library robotics can handle 30 mount, tape positioning, and demount cycles per hour for each drive.

Note that we did not size the amount of drives required for normal Tivoli Storage Manager housekeeping operations such as migration and database and storage pool backups. The assumption is that these operations can be performed in off peak periods and that the amount of data received on a daily basis by the Tivoli Storage Manager server is relatively low, in the order of tens of gigabytes. If the housekeeping window is small, the number of tape drives required might exceed those required for normal backup operations.

Should you use normal or WORM media? The answer to this last question depends on your interpretation of the regulations that govern the data being archived. If the data does not have to comply with any specific requirement such as non-erasable, non-rewriteable storage media, then you might easily use normal tape technology. On the other hand, if your application does require non-erasable, non-rewriteable storage media, then you might decide for a hierarchy composed of disk inside a DR550 followed by WORM tape devices such as:

- ▶ IBM TS1120 tape drives supporting both 100 GB and 500 GB WORM tape cartridges.
- ▶ IBM 3588 tape drives with LTO 3 400 GB native capacity and WORM media.

Sizing the Tivoli Storage Manager server

The Tivoli Storage Manager server has to be sized appropriately for storing the archive data and managing the data availability. That server must perform the following tasks:

- ▶ Receive archive data from the network and store it on storage pools.
- ▶ Migrate older data from disk to tape.
- ▶ Perform backups of the Tivoli Storage Manager database and the Tivoli Storage Manager primary storage pools.
- ▶ Satisfy recall requests, read data from storage media, and send it back to the client.

These tasks cause traffic on the Tivoli Storage Manager server — traffic in the sense that these tasks consume system resources such as CPU cycles and network and storage bandwidth. If more data is received, a more powerful server is required. The server should be sized to accommodate all these tasks running concurrently. In general, the amount of data archived on a daily basis and received from a Tivoli Storage Manager server used for data archival is relatively low, in the order of tens of gigabytes a day. These low to medium data rates can be satisfied by industry standard servers. Care should be taken only when you must handle large files, in the range of hundreds of megabytes, and if archive and retrieve performance is important.

For additional information, refer to the Tivoli Storage Manager sizing and planning questionnaire that can be found at:

<http://www-03.ibm.com/support/techdocs/atsmastr.nsf/585741c64201a45286256ccf00653ad4/3203fe67c4054f048625709f006f3471?OpenDocument>

Or see the IBM Redbook, *IBM Tivoli Storage Manager Implementation Guide*, SG24-5416, at: <http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg245416.html?Open>

8.2 Business continuity and disaster recovery considerations

In this section we illustrate the functions that Tivoli Storage Manager offers to protect data in the case of an event such as a hardware failure or disaster that make the Tivoli Storage Manager server and the storage unavailable.

Generally, archival data has to be stored for long periods of time. It differs in a fundamental way from backup data. Backups are a second copy of data that is available on a system. They are only used when the primary data gets corrupted or is lost for whatever reason, and they are produced on a regular basis, often at daily or weekly intervals. However, archival data is often the last valid copy. If a corruption or disaster strikes the Tivoli Storage Manager server and the server storage environment, the data can be lost. This data can be protected by using Tivoli Storage Manager functions that make copies of the archival data and metadata for onsite or offsite storage.

As illustrated in Figure 4-2 on page 75, the Tivoli Storage Manager server environment is made up of three main components:

- ▶ The Tivoli Storage Manager server application and the server hardware where it runs.
- ▶ The Tivoli Storage Manager server data base, which contains metadata on stored objects and their location.
- ▶ The Tivoli Storage Manager server primary storage pools, which contain the data that has been archived.

In the following sections we discuss various approaches, tools, and architectures to protect the three main components of a Tivoli Storage Manager environment.

8.2.1 Protecting the server and the database

Tivoli Storage Manager is a software application that runs on a server. As any other software application, it is subject to the availability of the server's hardware components such as CPU, memory, IO and network access. Redundant components are used to guarantee application availability and it is no different for Tivoli Storage Manager.

Examples of redundant components are:

- ▶ Multiple Host Bus Adapters (HBA) and multipathing software to access disk devices
- ▶ Multiple network cards with IP address failover
- ▶ Protected memory chips

Tivoli Storage Manager clustering and failover

If the primary server does fail completely, then the Tivoli Storage Manager server application can be restarted on a failover server. To do this, Tivoli Storage Manager must be configured appropriately. For example, the Tivoli Storage Manager product code must be installed on both the primary server and on the failover server, external storage devices must be accessible by both servers, and the Tivoli Storage Manager application's disks containing files such as database, database log, and disk storage pools must be switched over to the failover server.

This failover can be performed manually by the operator following an installation developed procedure that documents the steps. The procedure should be regularly tested to guarantee that it will work when required.

The failover process can also be automated using clustering software that monitors the availability of the Tivoli Storage Manager server process and Tivoli Storage Manager server resources and restarts the Tivoli Storage Manager server application on a failover server in the case of unavailability of the primary server. IBM supports multiple clustering solutions for the Tivoli Storage Manager server, such as HACMP for Tivoli Storage Manager server on AIX systems or Microsoft Cluster in a Windows environment. Other clustering solutions can be implemented for Tivoli Storage Manager using various clustering software products.

For more information, refer to the IBM Redbook, *IBM Tivoli Storage Manager in a Clustered Environment*, SG24-6679.

Connecting to the Tivoli Storage Manager server

Tivoli Storage Manager is a client server application where Tivoli Storage Manager clients and Tivoli Storage Manager administrators communicate with the Tivoli Storage Manager server over the network. If the network is not available, then the clients cannot communicate with the Tivoli Storage Manager server, and all Tivoli Storage Manager operations stop.

Standard network availability procedures also apply to the Tivoli Storage Manager server. We recommend that you have multiple LAN network interfaces on the Tivoli Storage Manager server and a software mechanism that can either load balance or failover the IP address from one interface to another.

We also recommend that you use IP symbolic names and use Domain Name System (DNS) address resolution to translate the symbolic addresses to IP numerical addresses. This simplifies management of a Tivoli Storage Manager environment, because the Tivoli Storage Manager server numerical address can easily be reconfigured in the DNS server instead of manually having to edit a potentially large number individual Tivoli Storage Manager client addresses.

When performing manual failover, you must remember to switch the Tivoli Storage Manager server IP address to the address of the new server. This can be done either by reconfiguring the old IP address on a network adapter in the new server or by switching DNS resolution of the symbolic IP address to the numerical address representing the new server.

Protecting the Tivoli Storage Manager server database

The Tivoli Storage Manager database contains information about the client data archived in your storage pools. The recovery log contains records of changes to the database. If you lose the recovery log, you lose the changes that have been made since the last database backup. If you lose the database, you lose indexing to your client data. You have several ways to protect this information:

- ▶ Mirror the database, or the recovery log, or both.
- ▶ Back up the database to media such as tape, other sequential devices, or Tivoli Storage Manager remote virtual volumes.

Tivoli Storage Manager software mirroring protects against hardware failure of the storage device that contains the Tivoli Storage Manager database, but it does not protect against logical errors such as operator errors in the Tivoli Storage Manager server database. Tivoli Storage Manager offers integrated software mirroring for the database and log volumes; up to three mirrors can be kept. When one database or log volume copy becomes unavailable, the Tivoli Storage Manager server will report an error and continue operating.

You can perform full and incremental Tivoli Storage Manager database backups to tape while the server is running and available to clients. There are two modes of backing up the Tivoli Storage Manager database:

- ▶ Normal mode allows you to recover to a point-in-time of the latest full or incremental backup only.
- ▶ Roll-forward mode allows you to recover to a point-in-time of the latest full or incremental backup or, with an intact recovery log, to the most current state.

With the server running in normal mode, the backup media can then be stored onsite or offsite and can be used to recover the database up to the point of the backup. You can run full or incremental backups as often as required to ensure that the database can be restored to an acceptable point-in-time.

You can provide even more complete protection if you specify roll-forward mode. With roll-forward mode and an intact recovery log, you can recover the database up to its most current state (the point at which the database was lost).

For the fastest recovery time and greatest availability of the database, mirror both the database and recovery log, and periodically back up the database. When operating in roll-forward mode, mirroring better ensures that you have an intact recovery log, which is necessary to restore the database to its most current state.

Backing up the Tivoli Storage Manager database is a simple operation. You can back up the database with full and incremental backups or by taking a snapshot of a specific point-in-time of the database; these are called snapshot database backups.

Multiple media types are supported for the backup of the Tivoli Storage Manager database. The requirements are that the media be managed as a sequential device class by Tivoli Storage Manager. Example of supported devices for the backup of the Tivoli Storage Manager database are tape, DVD, files on a disk storage or Tivoli Storage Manager virtual volumes, and volumes that are written in a separate Tivoli Storage Manager server.

Restriction: Virtual volumes are not supported in SSAM or in the DR550.

We recommend that you back up the Tivoli Storage Manager database at least once a day.

For additional information on managing Tivoli Storage Manager database backups, refer to the specific Tivoli Storage Manager server administration guide for your operating system platform, which can be found at:

<http://publib.boulder.ibm.com/infocenter/tivihelp/v1r1/index.jsp>

8.2.2 Protecting the Tivoli Storage Manager primary storage pools

Tivoli Storage Manager data is stored in storage pools, collections of storage devices with common characteristics. For more information on storage management, see 4.1.2, “Tivoli Storage Manager storage management” on page 82. This data can be protected by using Tivoli Storage Manager copy storage pools.

You can back up primary storage pools to copy storage pools to improve data availability. When you back up a primary storage pool, you create backup copies of client files that are stored in primary storage pools in copy storage pools. By using copy storage pools, you maintain multiple copies of files and reduce the potential for data loss due to media failure. If the primary file is not available or becomes corrupted, the server accesses and uses the duplicate file from a copy storage pool.

If data is lost or damaged, you can restore individual volumes or entire storage pools from the copy storage pools. The server automatically tries to access the file from a copy storage pool if the primary copy of the file cannot be obtained for one of the following reasons:

- ▶ The primary file copy has been previously marked damaged.
- ▶ The primary file is stored on a volume that UNAVAILABLE or DESTROYED.
- ▶ The primary file is stored on an offline volume.
- ▶ The primary file is located in a storage pool that is UNAVAILABLE, and the operation is for restore, retrieve, or recall of files to a user, or export of file data.

Primary storage pools should be backed frequently, for example, up each day, to the same copy storage pool. Figure 8-4 illustrates a sample Tivoli Storage Manager storage pool structure. We have three storage pools:

- ▶ *Diskpool* is where data is stored when it is received by the Tivoli Storage Manager server. Diskpool data migrates to tapepool when predefined utilization and age thresholds are exceeded.
- ▶ *Tapepool* is the next storage pool to diskpool, the next level in the storage hierarchy.
- ▶ *Copypool* contains copies of all data stored in both *diskpool* and *tapepool*.

Backups of primary storage pool data to the *copypool* copy storage pool are performed by running the Tivoli Storage Manager **backup storagepool** administrative command. In the specific case, you must run two **backup storagepool** commands, one to back up diskpool to copypool, and one to back up tapepool to copypool. Backing up to the same copy storage pool ensures that files do not have to be recopied after they have migrated to the next storage pool.

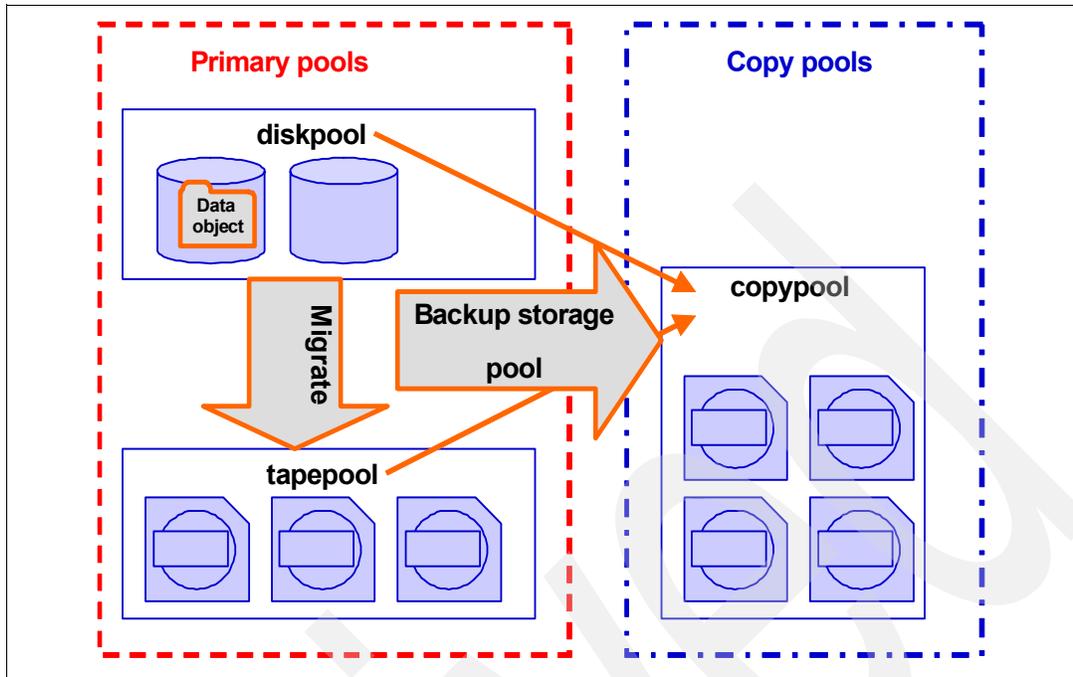


Figure 8-4 Tivoli Storage Manager backup storage pool structure

Because the backup copies are made incrementally, you can cancel the backup process. Reissuing the `backup storagepool` command lets the backup continue from the spot where it was canceled.

You can back up multiple primary storage pools to one copy storage pool. If multiple copies are necessary, you can also back up a primary storage pool to multiple copy storage pools. However, you should back up the entire primary storage pool hierarchy to the same copy storage pool for easier management of storage volumes.

You can set up a primary storage pool so that when a client backs up, archives, or migrates a file, the file is written to the primary storage pool and is simultaneously stored into each copy storage pool specified for the primary storage pool. This function can be used to create duplicate copies of data synchronously in environments where disk storagepool storage mirroring is not an option.

Use of the simultaneous write function is not intended to replace regular backups of storage pools. If you use the function to simultaneously write to copy storage pools, ensure that the copy of each primary storage pool is complete by regularly issuing the Tivoli Storage Manager commands to back up the primary storage pools.

For the best protection, primary storage pools should be backed up regularly, preferably each day. You can define Tivoli Storage Manager administrative schedules to begin backups of files in the primary storage pools on a regular basis.

8.2.3 Tivoli Storage Manager Disaster Recovery Manager (DRM)

We have discussed the requirement and methods to schedule regular Tivoli Storage Manager database and storage pool backups on a daily basis. We can send these backup volumes to an offsite location so that they can be used in the case of a disaster that makes the Tivoli Storage Manager server environment unusable.

Tivoli Storage Manager Disaster recovery is the process of restoring Tivoli Storage Manager operations in the event of a catastrophe. There are many aspects to consider related to the restoration, including facilities, equipment, personnel, supplies, customer services, and data. One of the most valuable business assets is the critical data that resides on the computer systems throughout the company, or in the case of archival data the data stored in the Tivoli Storage Manager server itself. The recovery of this data is a primary focus of the disaster recovery plan. Tivoli Storage Manager, along with the Tivoli Storage Manager Disaster Recovery Manager (DRM) function included in Tivoli Storage Manager Extended Edition, will assist you in the technical steps that you must perform to make your data available after a widespread failure.

Distributed data recovery restores data to workstations, application servers, and file servers in the event of data loss due to accidental erasure, media failures, sabotage, and natural disasters. It involves creating, managing, and recovering copies of distributed data. These copies should be taken off-site to minimize the chance that a disaster will destroy backup copies along with primary copies. Many data administrators choose to keep backup copies on-site also, to expedite recovery from smaller media failures.

Disaster recovery requires, at a minimum, creating copies of primary data. Many businesses and backup products stop here. To achieve a complete recovery solution for distributed data, several additional features must be considered, such as offsite media movement and rotation and documenting the Tivoli Storage Manager procedures required in case of a disaster recovery.

Tivoli Storage Manager DRM coordinates and automates the process of recovering from a disaster. It provides for off-site media management, automated restore of the Tivoli Storage Manager server, and managed client recovery. It complements the already implemented robust protection features of Tivoli Storage Manager and automates many already facilitated protection functions.

DRM automatically captures information required to recover the Tivoli Storage Manager server after a disaster. It assists in preparing a plan that allows recovery in the most expedient manner. This disaster recovery plan contains information, scripts, and procedures required to automate and facilitate server restoration and helps ensure quick recovery of your data after a disaster. DRM also manages and tracks the movement of off-site media to reduce the time required to recover in the event of a disaster. It is able to track media that are stored onsite, in-transit, or off-site in a vault, no matter whether it is a manual or electronic vault, therefore your data can be easily located if disaster strikes.

Figure 8-5 shows the DRM media cycle.

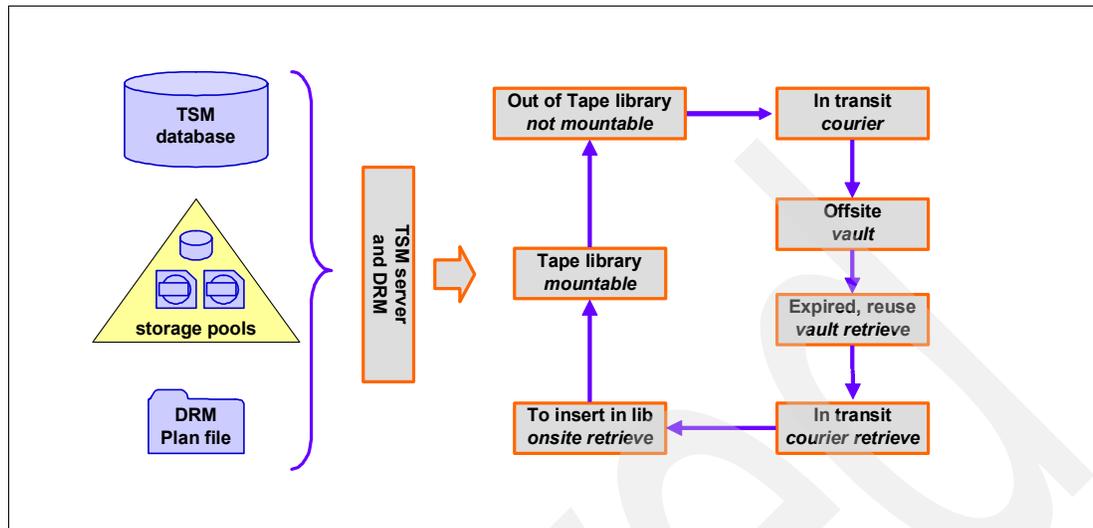


Figure 8-5 Tivoli Storage Manager DRM and offsite media flow

Client recovery information can also be captured by DRM. This information can be used to assist with identifying what clients must have recovered, in what order, and what is required to recover it, including data and media that is not managed by Tivoli Storage Manager. Client recovery is not considered in the context of the SSAM, because regular Tivoli Storage Manager backup and archive clients cannot store data in SSAM.

In a typical protected Tivoli Storage Manager environment, after each day's of clients storing data in the Tivoli Storage Manager server, the copy storage pools are also updated with the new data. Then, a server database backup is done. The newly generated volumes are sent to a safe location, and a recovery plan file is regenerated by DRM to make sure it includes the latest information. As data expires from the on-site pools, it also expires from the off-site pools and unnecessary database backups. Disaster Recovery Manager also tracks such media as they become empty so that you can report on free tapes that can be brought back on-site for reuse.

Volume tracking

Disaster Recovery Manager provides several levels of volume tracking. Disaster Recovery Manager volume management includes:

- ▶ Identifying which off-site volumes are required for a given recovery: Disaster Recovery Manager knows the volumes that are associated with each primary Tivoli Storage Manager server backup so that you can initiate a complete recovery of all storage pools, or only a partial recovery, depending on the extent of the disaster. You can also configure Disaster Recovery Manager to track volumes only from certain storage pools (this is useful, for example, to provide critical client nodes full off-site protection, and other, less-critical nodes, no off-site protection).
- ▶ Integrating with tape management systems: Because Disaster Recovery Manager is fully integrated with tape management, every time a new tape is created in the corresponding copy storage pools, it is automatically eligible for off-site movement.
- ▶ Recycling partially filled volumes: Off-site volumes are reclaimed just as on-site volumes are. Disaster Recovery Manager enables you to see which volumes have reached an empty state because of reclamation so that you can request them to be returned on-site. This feature is not applicable for WORM media pools, where space reclamation is not enabled.

- ▶ **Tracking off-site volumes:** This is one of Disaster Recovery Manager's strongest features. Disaster Recovery Manager manages tapes by assigning a special, predefined set of states to each off-site tape. Depending where the tape should be, there are two possible directions for a tape: from on-site to off-site and from off-site to on-site. The first starts during normal backup processing to save up-to-date data to the copy storage pool. The tapes pass through a number of states in their journey from the production tape library to the safe vault. Then, time elapses while the tape remains off-site, ready to be used for a restore in the event of a disaster. During this time, data is gradually expiring from the tape. When the tape finally reaches its reclamation threshold, it is reclaimed by normal processes. After it is empty, it moves in the reverse direction, that is, it is returned onsite for reuse. Again, with the use of WORM media and space reclamation turned off, the journey back on-site will only occur if a disaster recovery has to be performed.

To make the creation and maintenance of the server disaster recovery plan easier, the **prepare** command automatically queries the required information from the Tivoli Storage Manager server and creates the recovery plan file. The **prepare** command can be scheduled using the Tivoli Storage Manager central scheduling capabilities.

Auditable plan for the Tivoli Storage Manager server

The recovery plan file contains the information and procedures necessary to assist with the recovery of the Tivoli Storage Manager server. The information in the plan file includes site-specific server recovery instructions and information as defined by the administrator (for example, contact names and telephone numbers for important people and their backups).

Here is the sequence that is necessary to recover a Tivoli Storage Manager server:

1. List of Tivoli Storage Manager database backup and copy storage pool volumes required to perform the recovery (including the off-site location where the volumes reside)
2. Devices required to read the database backup and copy storage pool volumes
3. Space requirements for the Tivoli Storage Manager database and recovery log
4. Copy of Tivoli Storage Manager server options file, device configuration file, and volume history information file
5. Shell scripts (on UNIX) and Tivoli Storage Manager macros for performing server database recovery and primary storage pool recovery

Off-site recovery media management

Knowing the location of off-site recovery media is critical to the successful implementation of a disaster recovery management plan. The off-site recovery media management function provides:

- ▶ Determination of which database and copy storage pool volumes must be moved off-site and back on-site
- ▶ Automatic ejection of volumes from an automated library
- ▶ Tracking of the media location and state in the Tivoli Storage Manager database

This function allows database backup volumes and copy storage pool volumes to be treated as logical collections that are selected to move off-site for safekeeping and on-site for use. The reclamation of off-site volumes includes the capability to specify the number of days to retain a Tivoli Storage Manager database backup series. After the expiration interval is reached, the data on the media is no longer considered to be valid. The media can then be reused (or disposed of).

Figure 8-6 illustrates how your off-site data could be used to recover your environment. Note that V1 is the point in time requested; therefore, you cannot only rebuild the latest one, but also data from any specific point in time that you still have saved. The execution of the recovery scripts (which perform the Automatic Recovery Steps in the figure) starts after you have reinstalled the operating system and Tivoli Storage Manager server code on your replacement server hardware.

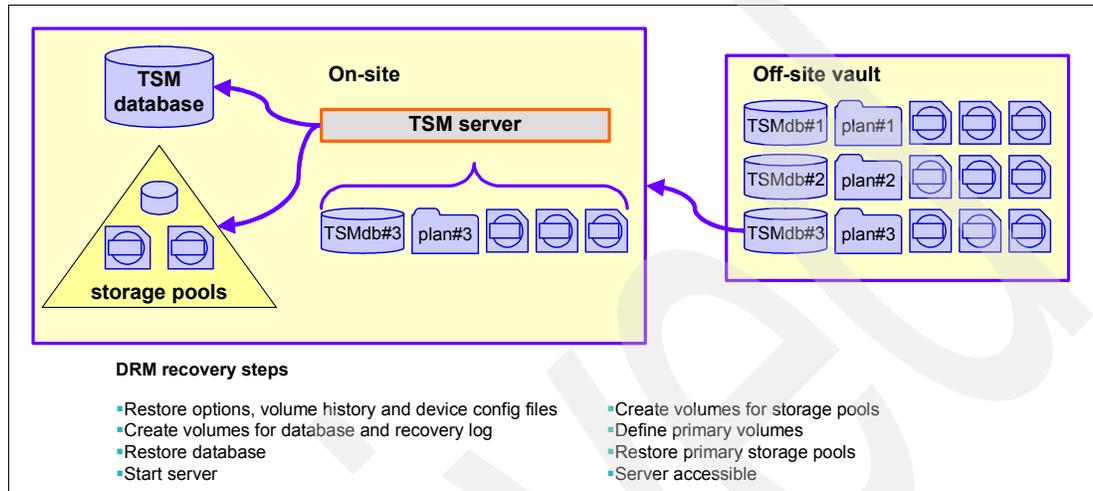


Figure 8-6 Restoring a Tivoli Storage Manager server with DRM

Additional disaster recovery issues

Disaster recovery goes far beyond simple technical measures. To have a fully operational and prepared environment, you must also pay attention to additional issues, such as those described in the following sections.

Hardware system requirements

Disaster Recovery Manager creates a recovery plan file based on the information and space allocation on the Tivoli Storage Manager production server machine. This means that you must evaluate whether to have a similar machine for off-site recovery and make the changes to fit the new environment.

Additional operating system recovery steps

Depending on the operating system on which Tivoli Storage Manager is installed, you might have to send special CD or tape images (for the specific OS recovery steps) to the off-site location. For example, this would be fully supported on an AIX machine by using the `mksysb` operating system command to produce a valid, bootable tape or DVD image of your present configuration.

Recovery testing

A recovery solution must be tested before it is actually required. A good approach is to create all documents, operating system tapes, special hardware requirements, and installation scripts, and send them to the off-site location labeled as a “Disaster Recovery starter kit.” Then, perform a complete recovery test once a year to ensure that the documents are accurate for recovery and incorporate any changes that were uncovered during your test.

Further information about disaster recovery concepts, and especially the DRM, can be found in the IBM Redbook, *IBM Tivoli Storage Management Concepts*, SG24-4877, available at:

<http://www.redbooks.ibm.com/abstracts/sg244877.html>

8.2.4 Sample high availability and disaster recovery configurations

A Tivoli Storage Manager or SSAM environment can be easily configured or upgraded for high availability and redundancy of components. We distinguish between high availability configurations and disaster recovery and vaulting. We define the following terms:

- ▶ *High availability* exists when you have a Tivoli Storage Manager server that can failover to a separate machine in the case that one machine fails.
- ▶ *Disaster recovery* relates to when the Tivoli Storage Manager server can be restarted at a remote location, on the assumption that the primary location is no longer available.
- ▶ *Vaulting* is the process of moving a copy of the data stored in Tivoli Storage Manager to a secure location, a location that should not be impacted by a disaster that makes the primary site unavailable.

A Tivoli Storage Manager server requires the following components and services:

- ▶ A server to run on with adequate system resources such as CPU and memory.
- ▶ Disk space to store the Tivoli Storage Manager database and configuration files.
- ▶ Storage space, such as disk and tape, to store the actual data.
- ▶ Access to the LAN network to receive data from the clients.

Local cluster configuration

The simplest high availability configuration is the classic cluster setup shown in Figure 8-7. In the diagram we see two servers called CPU#1 and CPU#3 connected to an external Storage Area Network (SAN). The Tivoli Storage Manager server code is installed on both servers. The SAN also connects to both disk and tape storage: we have external disk Disk#1 and tape library TapeLib#1; these storage devices must be accessible to both servers.

The Tivoli Storage Manager server TSMsRv#1 is active on CPU#1. Server instance TSMsRv#1 has its data on Disk#1 in the volume, or group of volumes, called TSM#1D-C1. TSM#1D-C1 contains the Tivoli Storage Manager control files, the database and log and all the disk storage pools. Tape library TapeLib#1 contains server TSMsRv#1 tape volumes, indicated as TSM#1T-C1.

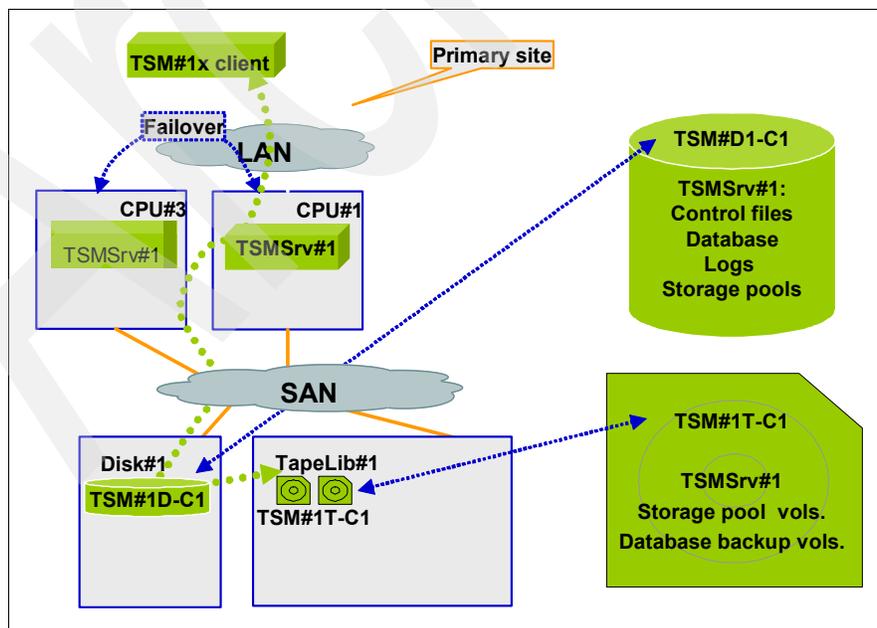


Figure 8-7 Tivoli Storage Manager sample local high availability setup

If the server called CPU#1 fails, TSMSrv#1 can fail over to CPU#3. The failover can be either manual or automated with clustering software. To accomplish the failover, the following actions must be performed, either by clustering software or manually by an operator:

1. Connect the storage subsystem volumes called TSM#1D-C1 to CPU#3 and make them accessible using appropriate operating system commands.
2. Ensure that the tape library and the volumes called TSM#1T-C1 are available and can be accessed by CPU#3.
3. Failover the TCP/IP address so that clients can find the TSMSrv#1 service when it is restarted on CPU#3.
4. Restart the TSMSrv#1 service on CPU#3.

This setup ensures that the Tivoli Storage Manager application can be restarted in the event that you lose access to server CPU#1. If you lose access to the external disk or tape storage devices, Tivoli Storage Manager will either function in degraded mode or will not be able to start, depending on the extent of the damage to the database and control files that are required for Tivoli Storage Manager to start up.

Stretched cluster configuration

In this second example we discuss the addition of a secondary site with redundant servers and storage devices. This is an extension to cluster configuration discussed in “Local cluster configuration” on page 198. The second site can be located at some distance from the first site, a distance that depends on the type of storage mirroring techniques you use.

Figure 8-8 schematically illustrates the layout of the two sites. The second site has SAN connected storage devices: a disk subsystem called Disk#2 and a tape library called TapeLib#2. The second site also contains a standby server called CPU#2, with the same operating system and Tivoli Storage Manager software levels installed.

TSMSrv#1 in normal conditions runs on CPU#1. It writes its data to local disk and tape. The disk storage can be replicated to the remote site using various techniques depending on operating system and disk storage subsystem type.

One option to replicate the data between the Disk#1 and Disk#2 disk storage subsystems is to use disk hardware mirroring functions such as either Metro Mirror or Global Mirror or SnapMirror, available on the IBM DS4000, DS6000 and DS8000 and N series families of disk storage devices and in the IBM SAN Volume Controller. A second option is to use software mirroring products running on CPU#1 and CPU#2.

You should replicate all Tivoli Storage Manager storage between the two sites, both the database, logs and control files and the storage pools. If this data is replicated synchronously or near synchronously the loss of data in the event of a disaster might be zero or small, limited to the time lag of an asynchronous replication solution such as Global Mirror. All the Tivoli Storage Manager data should be managed in one consistency group to ensure there is consistency between metadata and storage pool data in the secondary site.

For more information on disk storage mirroring solutions, refer to the IBM Redbook, *IBM TotalStorage Business Continuity Solutions Guide*, SG24-6547.

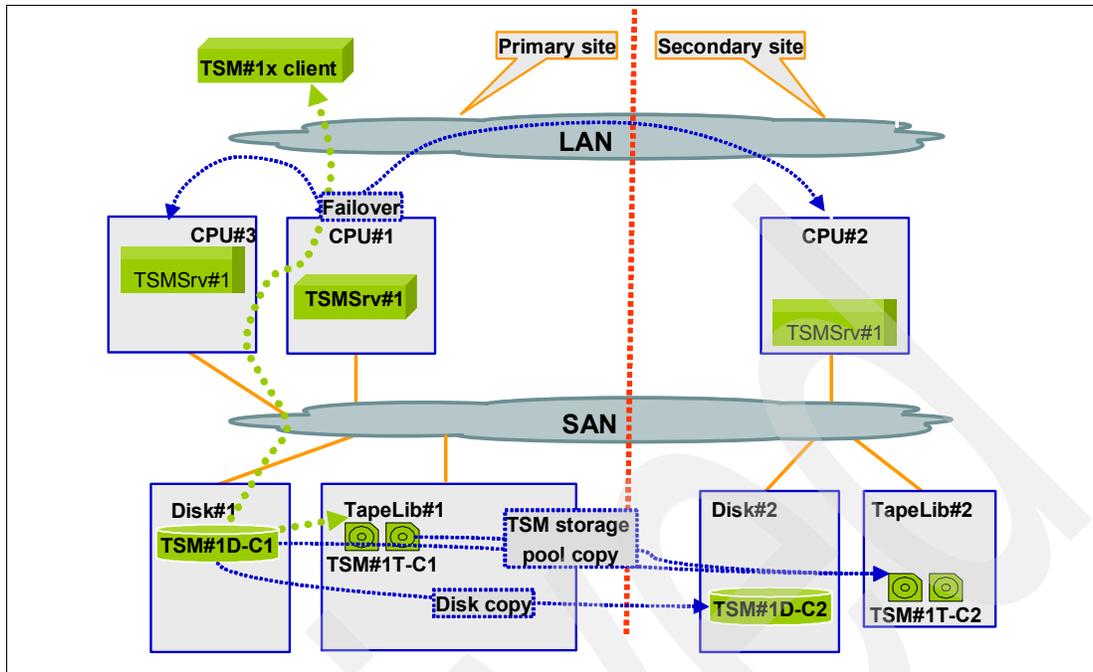


Figure 8-8 Stretched cluster configuration

The replication of data between TapeLib#1 and TapeLib#2 is performed by the TSM#1T-C1 using the server copy storage pools, discussed in “Protecting the Tivoli Storage Manager primary storage pools” on page 192. This type of replication requires SAN connectivity so that each server can access both tape libraries at the same time. In the example, CPU#1 has to access the tape drives in both TapeLib#1 and TapeLib#2, and to reach the drives in TapeLib#2 remote SAN connectivity is required.

Tivoli Storage Manager can be configured to copy data synchronously or asynchronously between primary and copy storage pools. Data must be copied between both the disk and tape primary storage pools, TSM#1D-C1 and TSM#1T-C1 primary, or C1 copy one, storage pools to the TSM#1T-C2 copy two storage pool.

In synchronous mode, the data is received from the network and written to both the primary and copy storage pools, and then a transaction complete status is issued to the client. The copy storage pool to use is configured as an attribute of the primary storage pool.

In asynchronous mode, the data is first written to the primary storage pools and subsequently copied to the copy storage pools. This copy is performed by running the backup storage pool command for all the primary storage pools. The command can be scheduled on a periodic basis, for example at daily intervals, by using the Tivoli Storage Manager scheduler.

We illustrate failover in a scenario where both primary site servers, CPU#1 and CPU#3, and all local storage devices, Disk#1 and TapeLib#1 are no longer available. To perform the failover of TSM#1T-C1 between CPU#1 and CPU#2, assuming Metro Mirroring is in use, you must carry out the following actions. These actions can either be performed automatically by clustering software or manually by an operator:

1. Suspend Metro Mirror on Disk#2 to make TSM TSM#1D-C2, the second copy of the target volumes accessible to CPU#3.
2. Connect the secondary site storage subsystem volumes called TSM#1D-C2 to CPU#3 and make them accessible to the operating system using specific operating system commands.

3. Ensure that the secondary site tape library, TsmLib#2, and the volumes TSM#1T-C2 are available and can be accessed by CPU#2.
4. Failover the TCP/IP address so that clients can find the TSMSrv#1 service when it is restarted on CPU#2.
5. Use Tivoli Storage Manager commands to mark all the primary volumes on the primary site as unavailable. These are the tape volumes in TapeLib#1 called TSM#1T-C1.
6. Ensure all data has been copied between the primary storage pools and the copy storage pools. This is especially important when asynchronous copying of data to the copy storage pool is used.
7. Restart the TSMSrv#1 service on CPU#2.

DR550 replication

The DR550 offers the Enhanced Remote Mirroring (ERM) option, a feature of the IBM DS4000 Storage Manager software. ERM is used for online, real-time replication of data between storage subsystems at different sites. It allows the secondary storage subsystem to take over responsibility for primary I/O operations. ERM supports:

Metro Mirroring for synchronous mirroring mode, for distances less than 100 miles and latency less than 10ms.

Global Mirroring is an asynchronous write mode that ensures that the write requests are carried out in the same order at the remote site and it is used at longer distances, typically greater than 100 miles.

Two site active-active configuration

The example shown in Figure 8-8 shows an active-active stretched cluster configurations between the two sites. Building on the previous scenario where we had TSMSrv#1 on CPU#1 in this case we have another server instance called TSMSrv#3 that runs on CPU#2. The TSMSrv#2 data is replicated between the two sites in the same way as for TSMSrv#1. This server instance can then be restarted on CPU#1.

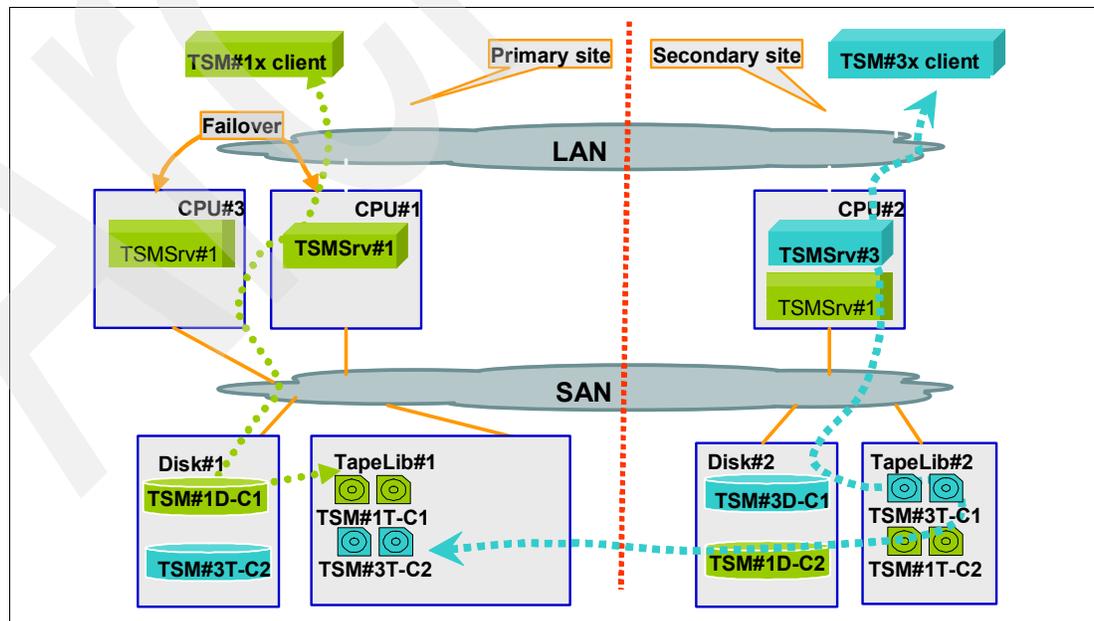


Figure 8-9 Active active stretched cluster

This allows you to build active-active configurations where the workload is partitioned, that means divided, between two sets of primary infrastructure, one in the primary site and one in the secondary site.

Writing to tape at a distance

Current technology tape drives such as the IBM TS1120 write to tape at sustained speeds of 100 MB/sec native transfer rate whereas the IBM 3588 tape drives have a native data rate of around 80 MB/sec. These data rates can be exceeded when writing compressible data; we can obtain up to 160 MB/sec on a 2 gbit Fibre Channel link. This is true for local devices, devices at a short distance from the server initiating the IO commands. For devices at larger distances, latency can become an issue and degrade performance noticeably.

Write acceleration, or fast write as it is sometimes called, is designed to mitigate the problem of the high latency of long distance networks. Write acceleration eliminates the time spent waiting for a target to tell the sender that it is ready to receive data. The idea is to send the data before receiving the ready signal, knowing that the ready signal will almost certainly arrive in due course. Data integrity is not jeopardized because the write is not assumed to have been successful until the final acknowledgement has been received anyway.

Figure 8-10 shows a standard write request, where each write operation is satisfied by two round trips, thus giving four times the one way link latency.

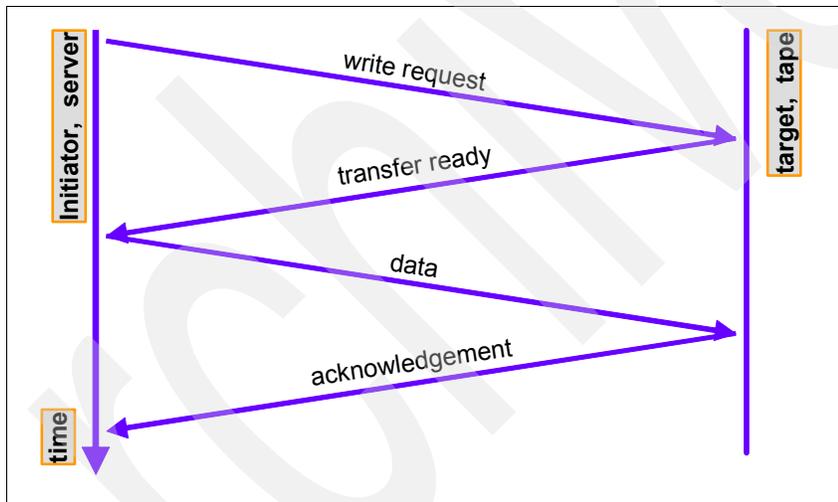


Figure 8-10 Standard write requests

Figure 8-11 shows write acceleration implemented in SAN switch hardware. The SAN switch can spoof, or issue of its own accord, a transfer ready command, thus avoiding the latency of one round trip. This improves write performance.

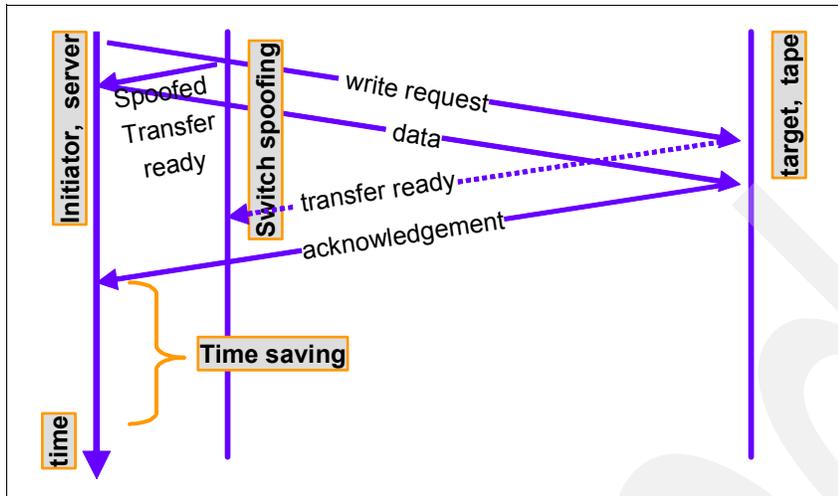


Figure 8-11 Accelerated write request

Tape acceleration implementations offer both transfer ready and acknowledgement spoofing, and this allows for good benefits in write performance. Refer to the IBM Redbook, *IBM TotalStorage: Introduction to SAN Routing*, SG24-7119 for additional information on tape write acceleration.

The performance benefit, though, is limited and experience shows that at distances of around 150 KM the data rate of high end tape drives drops to around 10-20 MB/sec, even given unlimited bandwidth. The disadvantage of the slow data rates is that tape cartridges have to be mounted longer to write the same amount of data: writing 100 MB/sec gives a data rate 360 GB/hour, but with a data rate of 20 MB/sec we only get 72 GB/hour, therefore, we require 5 hours to write 360 GB. With a degradation factor of five tape mount times are multiplied by 5. Therefore, you will require 5 times more tape drives, and this reflects also on the primary site: if data is written to tape at 20 MB/sec, it is read at the same speed.

To overcome this physical limitation, IBM offers the TS7510 Centralized Virtual Tape (CVT) product. TS7510 is a device that emulates tapes and writes the data to disk. This offers two advantages: the first is that it emulates tape drives and uses disk devices: it can emulate a large number of tape devices allowing for high levels of parallelization of operations, you are no longer constrained by few tape drives. The data can be sent slowly and tape start stop operations, bad for performance, are no longer an issue because disk drives do not suffer from such problems. The data can be replicated remotely to virtual tape in a TS7510. The TS7510 can later move the data to real, physical, tape drives with the TS7510 export function. The export is performed locally at the remote site and does not suffer from distance related latency problems.

When exporting a virtual tape to a physical tape, the first thing to do is to have Tivoli Storage Manager eject the virtual tape. When the virtual tape is ejected, it is automatically moved to a section of the TS7510 called the Virtual Vault. Auto Archive is an automatic export performed by TS7510 at the time Tivoli Storage Manager ejects the virtual tape. Options for Auto Archive are Copy and Move, and the tape can be moved to the I/E slots after this operation has finished.

For more information, refer to IBM Redbook, *IBM Virtualization Engine TS7510: Tape Virtualization for Open Systems Servers*, SG24-7189.

8.3 SSAM API essentials

The SSAM Application Programming Interface (API) allows applications to store data directly in SSAM without passing through a file system.

8.3.1 Programming to the SSAM API

In this section we provide basic information about the API that can be used by application architects and programmers who are creating applications that use the API. Refer to the manual, *IBM Tivoli Storage Manager Using the Application Program Interface*, GC32-0793, for additional information on the API. You should also refer to the Tivoli Storage Manager server and client manuals for additional information.

The API enables an application client to use storage management functions. The API includes function calls that you can use in an application to perform the following operations:

- ▶ Start or end a session
- ▶ Assign management classes to objects before they are stored on a server
- ▶ Back up or archive objects to a server
- ▶ Restore or retrieve objects from a server
- ▶ Query the server for information about stored objects
- ▶ Manage file spaces.
- ▶ Send retention events.

SSAM is specially configured to serve as an archival device for regulatory compliance purposes. SSAM cannot be used for backups or hierarchical storage management. Attempts to perform these functions via API calls will result in an error condition. Only archival or retrieval operations are permitted. Consequently, only an archive copy group can be utilized with event-based retention, and you must ensure that such a copy group exists on your SSAM server.

IBM System Storage Archive Manager is used in conjunction with external document management or records management applications. These applications perform the functions of record selection and categorization, and use the SSAM server, via the API, as a protected storage device.

SSAM imposes rigid controls on object expiration. After an object has been archived, it cannot be deleted. It can only expire. Nor can the expiration time be reduced, after it is set. See 4.3, “System Storage Archive Manager” on page 92 for more information.

Application design strategies

Because of the imposed controls, application design for data retention usually follows one of two alternative strategies:

- ▶ Use the chronological retention capability of SSAM and let it manage object expiration.
- ▶ Manage retention times and object expiration within the data management application.

The first strategy is the simplest to code and offers the additional benefit that the application does not require its own database. Record retrieval can be accomplished by querying the SSAM server database to obtain a list of objects meeting the selection criteria, then retrieving objects from that list. The primary disadvantage is that after a retention policy has been assigned to an object, it cannot be made shorter. Retention periods can be selectively extended by placing a hold event against objects, then releasing the hold at an appropriate time.

The second strategy retains full control of archival duration within the data management application by setting a minimal retention period in the SSAM archive copy group (RETVER setting), and controlling retention by use of *activate*, *hold*, and *release* events. This strategy is more complex for the application developer because the data management application is fully responsible for managing the retention period. This implies the requirement for an application-managed local database to keep track of the expiration policy for each object.

With regard to this local database, there is an important exception to the practice in using the API which recommends against keeping a local record of the Tivoli Storage Manager object identifier. This recommendation is intended to guard against possible Tivoli Storage Manager object identifier changes that can result from a database export-import operation. Because the import-export capability is disabled in IBM System Storage Archive Manager, the recommended practice does not apply and you might wish to keep the Tivoli Storage Manager object identifier in your local database to allow more efficient object retrieval operations.

Multiple clients using the same SSAM server

While it would simplify your code architecture if SSAM could be reserved for your application's exclusive use, that situation cannot be guaranteed. Most organizations that invest in SSAM tend to use its full range of capabilities. Therefore, your application client is likely going to share use of the SSAM infrastructure with other SSAM Archive clients. You should consider this fact in your design, as it will influence decisions about the default node name, to be used by your application, and possibly the storage policies and hierarchy on the Tivoli Storage Manager server.

The other major use of API Clients is for data retention. Specialized clients must be written using the API in order to select and archive data having retention and protection requirements.

Server considerations

One consideration is the possibility of multiple server instances running on a single system. This fully supported configuration is made possible by specifying different TCP listening ports in the SSAM server's `dsmserv.opt` file. Your application should make some provision for this possibility. The standard Tivoli Storage Manager Backup/Archive client utilizes the `dsm.sys` or `dsm.opt` file to specify server TCP addresses and ports. You can utilize these configuration files or code appropriate values into your `dsmSetup` or `dsmlnitEx` API calls. We suggest that, if you use a `dsm.opt` file, that you place it in a location other than the normal Tivoli Storage Manager Backup/Archive client's location.

Client node considerations

The Tivoli Storage Manager server recognizes each of its clients as a *node*. Therefore, in its simplest form, the node name is equivalent to the client name, which in turn is the host name. In the absence of a specified node name, a standard Tivoli Storage Manager Backup/Archive client will by default utilize the system host name as its node name. However, this default behavior is only one possible option. The same host can identify itself to the SSAM server by any number of different node names. Alternatively, more than one machine can interact with the SSAM server using the same node name.

There is one other thing the application developer has to know. Any node name used must be registered on the Tivoli Storage Manager server before it can successfully establish a session with that server. The server administrator typically performs this action on request, and must know at a minimum what node name and initial password to use.

Logical storage

The most significant advantage of SSAM, from the perspective of the storage application developer, is the way it abstracts the details of storage devices, hardware, into a logical construct that can be used by a relatively small set of API calls. Thus, the developer is free to concentrate on the application's functions without having to fuss over device drivers, and so on. In one sense, you could think of the Tivoli Storage Manager server as a sort of universal device driver that allows your application access to literally hundreds of storage device types, including disk arrays, tape libraries, optical jukeboxes, and so on.

The following paragraphs describe the basic logical structure your application code will be dealing with and illustrate how to set up such a structure to satisfy your particular requirements. Each of the elements described in the following sections can be thought of as a container for one or more of the elements immediately following.

Policy domain

The policy domain is the base element of the logical storage structure. An SSAM server can have many policy domains. The policy domain provides a convenient way to associate a Node with the appropriate data management policy set. Consequently, each node belongs to one, and only one, Policy Domain. This assignment is important because the domain assignment determines the policies the node will be subject to. It is normally a one-time assignment, although it can be changed if required.

Restriction: On an SSAM server configured for archive retention protection, a node's domain assignment can not be changed after data has been archived.

When a node name is registered, it can optionally be assigned to a specified Policy Domain. In the absence of a specified Domain, assignment will default to the preconfigured STANDARD Policy Domain. If your application requires its nodes to be assigned to a particular Policy Domain, this fact should be included in the installation documentation, and communicated to the Tivoli Storage Manager server administrator along with the other node registration information.

It is possible to obtain the Policy Domain information applicable to a session via the `dsmQuerySessInfo` call, and not a bad idea to check if you are using other than the standard defaults.

Policy set

Each Policy Domain has one active Policy Set. This set of policies determines how the Tivoli Storage Manager Server will handle data from nodes belonging to the Domain. A Policy Domain might contain any number of additional policy sets, but these will not be active. Think of any additional policy sets as a scratch pad used for development. Because it is not permitted to modify the active policy set, the only way to make changes is to make a copy of the active policy set, which will NOT be active, modify the copy, then validate and activate the newly-modified policy set. Only one Policy Set can be active at any given point in time, and this cannot be changed using the API calls, it can only be changed by an administrator on the SSAM Server.

Management class

The policy set contains one or more management classes. One management class must be designated as the default. The management class is used to specify the retention policies for a particular kind of data. In the absence of other instructions, data will be assigned to the default management class. However, SSAM supports multiple management classes within the same Policy Set. If your application requires special handling of its data, you might want special management classes established for this data. These special management classes

must be created on the Tivoli Storage Manager Server by an authorized administrator. You cannot perform this function within your API client, therefore it will be necessary for your product installation documentation to describe what is required.

Copy group

The management class contains either a backup copy group, or an archive copy group, or both. Only one copy group of each type can be defined for each management class, and the name for any copy group is *standard*. For your purposes, it is necessary only to know that the appropriate type of copy group must exist within the management class you are using, or your intended operation fails. This is not something that can be created by an API client, therefore unless you intend to use the default standard management class, you are required to specify the types of copy group your application requires in the installation documentation.

Note that if you intend to develop an application using event-based retention, you must have an archive copy group with the RETInit parameter set to EVENT. This is not the default, and your set-up documentation for the SSAM server must specify this.

Object naming and organization

Tivoli Storage Manager was originally developed to capture backups of information stored on computers. The internal database schema is consequently optimized for efficient storage and retrieval of data objects coming from a relatively large number of hosts or nodes with fairly typical file system structures.

Data organization options

Tivoli Storage Manager provides four basic levels of data organization:

- ▶ By node
- ▶ By filespace
- ▶ By high level name
- ▶ By low level name

Experience shows that the best Tivoli Storage Manager database performance is obtained when certain guidelines are followed. In a traditional backup application these guidelines are more-or-less followed by default because the filespace, high level name, and low level name typically follow the directory structure of the hosts being backed up. But other applications might require more attention paid to these organizational elements. As a general rule, database performance is best when the logical tree structure is balanced, not excessively wide or deep.

Organization by node

Each object backed up or archived on a Tivoli Storage Manager server is associated with a registered node name. Although it is possible to design an application using only one node name, we do not recommend this in most circumstances. Restoration or retrieval operation performance can degrade significantly for a node after the number of objects per node exceeds a few million. Tivoli Storage Manager can store five hundred million objects or more per instance and established best practices show that 100-200 nodes are a good maximum figure for a single Tivoli Storage Manager instance.

Organization by filespace

The first component of each object name is the filespace. Each filespace is associated with a specific node. Therefore, if you have ten nodes, each with a root filespace, these are treated as ten separate filespace, not one-even though they share the same name. Tivoli Storage Manager is optimized for database searches on the filespace. In a traditional backup application, the filespace corresponds with a single filesystem or logical drive. However, this association is arbitrary and your application can define filespace that have nothing to do with

physical storage architecture. For best performance, limit the number of filesystem names to less than 100 per node.

Organization by high level name

The high level name corresponds to the full path (excluding the filesystem name) in a traditional backup application. It is unique in that multiple organizational levels are possible, just as in a directory structure. The recommended rule of thumb is to create a new level of structure for every 256 entries. To illustrate, a filesystem named /fsname should have no more than 256 entries within it, for example /fsname/dir1, dir2 ...dir256. Then /dir1 could have 256 subentries, and so on.

Whatever you do, avoid the temptation to use an identical high level name for all objects from one node. Your structure should also avoid adding levels that will have only one sublevel entry. Both of these practices will adversely impact database performance.

Organization by low level name

In a traditional backup application, the low level name corresponds to an actual file name. This is the finest granularity possible within Tivoli Storage Manager, regardless of the application type.

Client sessions

Client sessions represent logical connections between client and server (Figure 8-12).

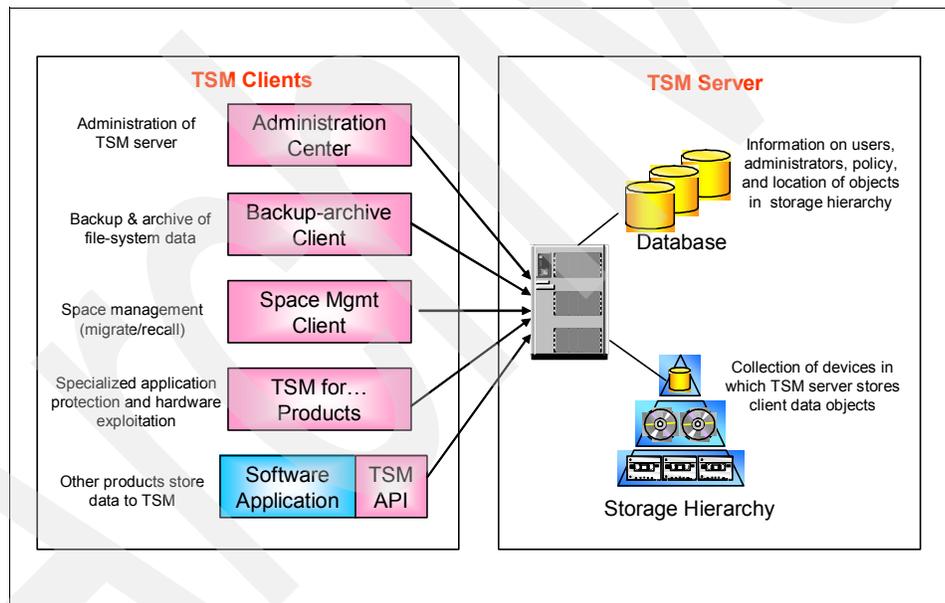


Figure 8-12 Client sessions

Random access and sequential access volumes

Tivoli Storage Manager recognizes two basic categories of media volumes. Random access media volumes are of device class DISK. Not only can these volumes be accessed at random locations, they can also be accessed simultaneously by multiple client sessions. The other category is sequential access. The various device classes associated with magnetic tape media are sequential access types. But the FILE device class, which resides on a disk device, is also sequential access. Sequential access volumes can only be accessed by one client session at a time.

Consequently it is vitally important that your application terminates its sessions promptly after performing an operation. Otherwise the possibility arises of a self-created deadlock situation.

Transactions

All work in Tivoli Storage Manager is performed in transactions. There is some overhead involved in transaction creation, about one real second per transaction, as a rule-of-thumb, therefore, for best application performance your design must attempt to minimize the total number of transactions used to move data. Tivoli Storage Manager provides the capability to aggregate multiple objects into a single transaction for transmission to the Tivoli Storage Manager server.

Questions concerning the application developer are, how to select objects for aggregation, and when to send these objects.

The primary selection criterion for aggregation is the management class. That is, all objects in an aggregate must be bound to the same management class. Therefore, before initiating a transmission session, you might want to segregate the pool of objects by management class so that appropriate aggregates can be assembled.

Timing is a function of the type of application. For applications that conduct scheduled backups or archival on a relatively infrequent basis, it is a safe assumption that these objects will reside on disk and can be accessed, categorized, and transmitted in one session. If your application will be used to back up or archive sporadically generated objects on a demand basis, then some local caching mechanism might be advisable to capture a reasonable number of objects before sending. Sending objects one per transaction as they arrive is not recommended as it will generally result in unacceptable throughput performance. In such cases, a cache usage threshold can be used to trigger session initiation and data transmission.

Note that in neither case do we recommend maintaining a continuously open session, primarily due to the possibility of volume contention discussed previously. Sessions should be started when you are ready to transmit or receive data, and terminated when the transmission is complete.

8.3.2 Application architectures

In this section, we discuss basic architectural design using the SSAM API. In each of the architectural diagrams provided in the figures, DMS refers to a generic Document Management System, not any specific product. Any resemblance of this label to any actual software product name is unintentional.

Single-client architecture

The single-client architecture, illustrated in Figure 8-13, features an API client on one server-class system, which manages data from multiple subsidiary hosts. This API client in turn communicates with a Tivoli Storage Manager server to back up or archive data. One advantage of this approach is simplicity of installation and maintenance. It is also advantageous for an application that maintains its own database of objects backed up or archived, especially when those objects could come from multiple host locations. The primary disadvantage of this architecture is inefficient network utilization. Each object transferred to Tivoli Storage Manager server storage might have to cross the network twice, once from the original source host to the API client host, then a second send from the client to the Tivoli Storage Manager server.

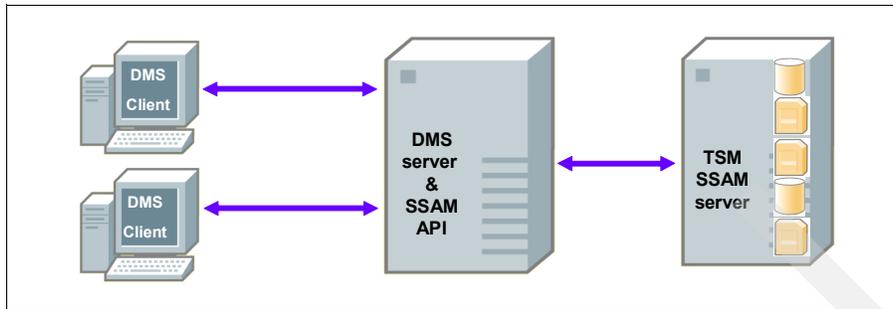


Figure 8-13 Single-client architecture

Multi-client architecture

The multi-client architecture, illustrated in Figure 8-14, features an API client on each host having data that will be backed up or archived to Tivoli Storage Manager server storage. This approach has the advantage of usually simpler API client design and better network utilization. However, it has some disadvantages as well. Maintenance workload will be higher due to the relatively larger number of API clients. If a centralized object repository is to be maintained other than the Tivoli Storage Manager internal database, this will be more complicated too with this approach.

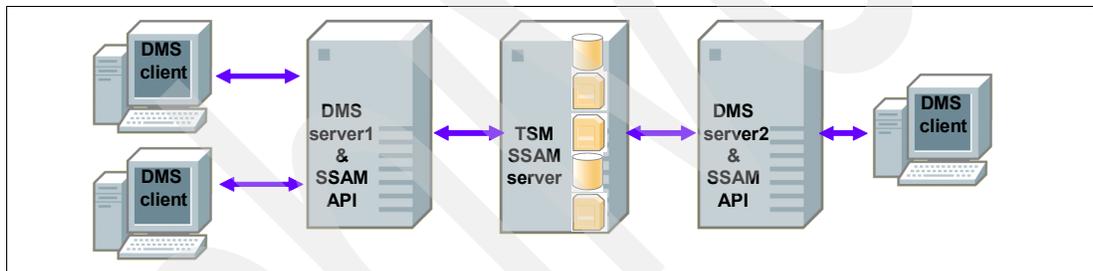


Figure 8-14 Multi-client architecture

Multi-server architecture

In very large environments the total number of objects to be stored might exceed the capacity of a single Tivoli Storage Manager instance. When this happens, multiple Tivoli Storage Manager servers can be installed and interconnected via a built in capability known as Enterprise Administration or the new Administration client introduced in Tivoli Storage Manager 5.3. In these circumstances two architectural design approaches can be taken. In multiple-client architectures, the clients can simply be distributed among the several servers. Each client is configured to access its single assigned server, usually by modifying the `dsm.opt` configuration file.

But in single-client architectures, the sole client must access all the servers and share the workload among them. It must be designed to:

- ▶ Be aware of the servers it can access.
- ▶ Choose the appropriate server for the intended operation.
- ▶ Either maintain awareness of which server has specific data (for retrieval purposes), or accept the performance implications of querying multiple servers for object retrieval.

This multi-server architecture is depicted in Figure 8-15.

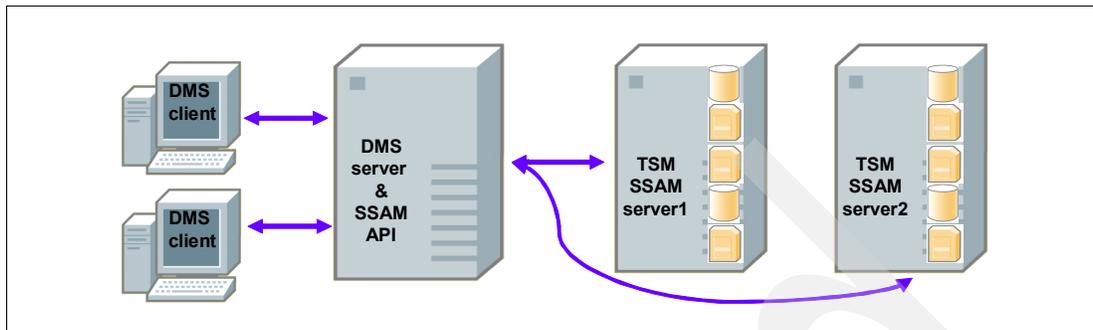


Figure 8-15 Multiple server architecture

Client-managed replication architecture

Some applications require a very high standard of availability, even in the event of a disaster. In these cases one design approach is to keep separate, independent copies of the same data on Tivoli Storage Manager servers in different geographical locations. While Tivoli Storage Manager has the capability to create remote copies, these do not satisfy certain regulatory requirements, therefore, in some data retention situations it might be necessary to use client-managed replication.

This is done by performing separate writes to two Tivoli Storage Manager servers. This in itself is not difficult. The challenge lies in keeping the two servers' contents synchronized. Your application must perform the appropriate error handling to ensure data consistency between the two Tivoli Storage Manager servers. Figure 8-16 illustrates this architecture.

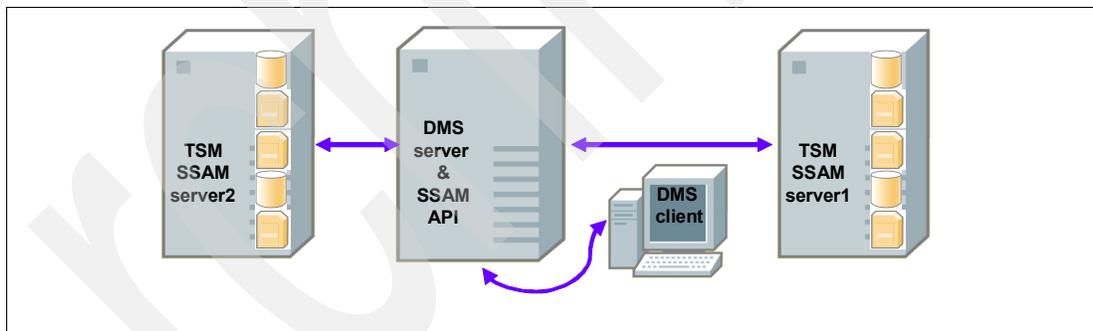


Figure 8-16 Client-managed replication

8.4 Using SSAM archive client for files

In this section we illustrate the use of the command line archive client to store files into SSAM. The command line client can be easily incorporated into scripts, and this makes integrating SSAM into data retention programs and procedures very simple.

Our environment consists of an SSAM client and server on the same machine. We have created the SSAM server environment with the commands shown in Example 8-1.

Example 8-1 SSAM commands used to set up environment

```
def domain do_ret
def policyset do_ret ps_ret
def mgmt do_ret ps_ret mc_event
def copy do_ret ps_ret mc_event t=a dest=archivepool retinit=event retver=3
retmin=2
def mgmt do_ret ps_ret mc_chrono
def copy do_ret ps_ret mc_chrono t=a dest=archivepool retinit=creation retver=3
assign defmg do_ret ps_ret mc_chrono
activate policyset do_ret ps_ret
reg node ret ret do=do_ret
```

We have created a policy domain called **do_ret**. This domain contains to management classes called **mc_event** and **mc_chrono**.

Management class **mc_event** is used for event based retention, **retinit=event** and has a minimum retention of 2 days, **retmin=2** and a retention of 3 days after the event has occurred, **retver=3**.

Management class **mc_chrono** is used for chronological retention; it will keep the object for three days after it has been stored in SSAM, **retver=3**.

We then created a node called **ret** and assigned it to policy domain **do_ret**.

Tip: Before you start archive tests in an SSAM retention protected server, we suggest that you use management classes with short expiration periods, because data archived even for tests cannot be deleted before it reaches its expiration date.

Before starting, we customize the SSAM archive client option file, whose default name is `dsm.opt`, and add or update the following two statements:

```
NODENAME ret
ENABLEARCHIVERETENTIONProtection yes
```

The `nodename` statement makes the client present himself to the SSAM server with the name **ret**. The client could be moved to a different machine with a different host name and still find its files in the Tivoli Storage Manager server. The second statement enables the client to use archive retention protection.

After this setup we are ready to archive our first file.

8.4.1 Archiving files with chronological retention

We first show an example of using the SSAM archive client to archive files using chronological retention, retention managed by Tivoli Storage Manager. We also show the use of the hold and release function to extend the lifetime of an object.

The first and simplest example is to archive a file, `G:\file5.txt`, using chronological retention; we will use the management class called **mc_chrono**. After archiving the file, we want to delete it from the disk. To do this we launch an SSAM archive client, **dsmc**, with the following options:

```
dsmc archive G:\file5.txt -archmc=mc_chrono -delete -desc='PROJECT9'
```

After successful completion of this command, the file has been archived and deleted from disk. The file has been assigned mc_chrono that has a 3 day retention value. We archive a second file:

```
dsmc archive G:\file7.txt -archmc=mc_chrono -delete -desc='PROJECT9'
```

To query the files that have been archived to Tivoli Storage Manager, you can issue the query archive command as follows, either for an individual file or for a group of files:

```
dsmc q archive G:\file*.txt -desc='PROJECT4'
```

Size	Archive	Date - Time	File - Expires on - Description
4 B	04/02/2006	03:21:29	\\tarella\g\$\file5.txt 04/05/2006 PROJECT9
4 B	04/02/2006	03:21:59	\\tarella\g\$\file7.txt 04/05/2006 PROJECT9

Assume that after a day, you determine that the file must be retained until further notice. In other words, you no longer want the file to expire on the planned expiration date. We can do this by issuing a hold event to the specific file with the dsmc SSAM command line client:

```
dsmc set event G:\file7.txt -type=hold
```

We can verify that the object was held by issuing the following query:

```
dsmc q archive G:\file*.txt -detail -desc='PROJECT9'
```

Size	Archive	Date - Time	File - Expires on - Description
4 B	04/02/2006	03:21:29	\\tarella\g\$\file5.txt 04/05/2006 PROJECT9
RetInit:STARTED ObjHeld:NO Modified: 04/02/2006 01:18:08 Created: 04/02/2006 03:20:40			
4 B	04/02/2006	03:21:59	\\tarella\g\$\file7.txt 04/05/2006 PROJECT9
RetInit:STARTED ObjHeld:YES Modified: 04/02/2006 01:18:08 Created: 04/02/2006 03:20:43			

To restore file7.txt to disk without renaming it, you can issue the following command:

```
dsmc retrieve G:\file7.txt
```

SSAM must be notified when the archived file that was previously held is no longer required. To notify SSAM, we issue a release command as shown:

```
dsmc set event G:\file2.txt -type=release
```

It is important to release the file so that it can be expired by normal Tivoli Storage Manager expiration processing.

8.4.2 Archiving files for event based retention

The second mode of managing retention is through event-based retention, where retention is initiated by the application: The application is responsible for starting object expiration.

We start by archiving two files to SSAM, G:\filea.txt and G:\fileb.txt, using management class mc_event:

```
dsmc archive G:\filea.txt -archmc=mc_event -delete -desc='PROJECTret'  
dsmc archive G:\fileb.txt -archmc=mc_event -delete -desc='PROJECTret'
```

To verify that the objects were archived and check the status, we use the query archive command:

```
dsmc q archive "G:\file*.txt" -detail
```

```
Size Archive Date - Time File - Expires on - Description
----
4 B 04/02/2006 21:50:19 \\tarella\g$\filea.txt Never 'PROJECTret'
RetInit:PENDING ObjHeld:NO Modified: 04/02/2006 01:18:08 Created: 04/02/2006
01:18:37
4 B 04/02/2006 21:50:49 \\tarella\g$\fileb.txt Never 'PROJECTret'
RetInit:PENDING ObjHeld:NO Modified: 04/02/2006 01:18:08 Created: 04/02/2006
02:35:23
```

We notice that both files are in the pending status because expiration has not yet been initiated for them. The *File - Expires on* field is set to **never**. To initiate the retention clock, we must issue the activate retention event for each file. If the **activateretention** event is not issued for a file, then the file will never be expired. In the example we issue the activateretention event to g:\fileb.txt.

```
dsmc set event G:\fileb.txt -type=Activateretention
```

After issuing the **set event** command, we use the query archive command again to verify the new status of g:\fileb.txt.

```
C:\TSM\baclient>dsmc q archive "G:\file*.txt" -detail
```

```
Size Archive Date - Time File - Expires on - Description
----
4 B 04/02/2006 21:50:19 \\tarella\g$\filea.txt Never 'PROJECTret'
RetInit:PENDING ObjHeld:NO Modified: 04/02/2006 01:18:08 Created: 04/02/2006
01:18:37
4 B 04/02/2006 21:50:49 \\tarella\g$\fileb.txt 04/05/2006 'PROJECTret'
RetInit:STARTED ObjHeld:NO Modified: 04/02/2006 01:18:08 Created: 04/02/2006
02:35:23
```

We notice that file g:\fileb.txt now has an expiration date. This expiration date is calculated as the maximum value between the retinit and retmin parameters of the archive copygroup associated with the mc_event management class.

It is the application's responsibility to initiate retention for the files using the activateretention event. If the application does not issue the event, the file will never be expired from SSAM storage.

The application can also issue the hold and release events as long as the file has not been expired. The hold and release commands work the same way as was illustrated in the chronological retention section.

8.4.3 SSAM and SnapLock best practices

One of our first suggestions for a retention-managed environment is to attempt to group data as much as possible by retention date to avoid SSAM SnapLock volume reclamation processing when part of a volume's data expires and the volume becomes available for reclamation because it has reached the reclamation period start date.

If you are not using SSAM deletion hold or event-based retention, reclamation probably will not be an issue because all files on a volume will expire by the first SSAM expiration run after the volume reaches the beginning of its reclamation period.

If you have two sets of data of the same size, one that expires after one year and a second one that expires after 10 years, and these are stored on the same SnapLock volumes, then you will experience inefficiencies in space usage: You will create volumes that have 50% of one year data and 50% of 10 year data. For the first year, space utilization efficiency will be 100% and for the next nine years, it will be 50% because the one year data has expired but the SSAM volume will not have reached its reclamation period.

If SSAM event-based retention or deletion hold functions are in use, it is not possible to estimate data retention requirements from SSAM management class parameters because retention is now effectively controlled by the content management application. In this case, we suggest you engage the application owners to understand data lifecycle requirements.

Data initially stored on an SSAM volume with SnapLock enabled can be moved or copied to a non-SnapLock volume. In this case, the moved or copied data loses the SnapLock hardware WORM protection and could be inadvertently or intentionally deleted. You must ensure that the data is stored on appropriate media to meet your legal requirements.

Tip: We do not recommend that you store data with less than three months retention period remaining on SnapLock protected volumes. For retention periods shorter than three months, evaluate other storage alternatives.

SnapLock WORM support is only enabled for SnapLock SSAM storage pools. If you define a Threshold storage pool and specify directories in the associated Tivoli Storage Manager device classes, which are SnapLock directories, your data will be in standard files, not WORM files. Therefore, you must ensure the storage pool definition has the parameter RECLAMATIONTYPE=SNAPLOCK set.

A SnapLock storage pool can only be defined with a FILE device class in which all directories are SnapLock directories. After a device class is associated with a SnapLock storage pool, updates to the directory structure must only be to SnapLock directories.

A file device class can only be shared between SnapLock storage pools or threshold storage pools: You should ensure that all storage pools that use the N series SnapLock device class have the parameter RECLAMATIONTYPE=SNAPLOCK set.

The SnapLock WORM function is only enabled for SnapLock storage pools. A customer might configure a Threshold storage pool pointing to a SnapLock file system, but the WORM function is not enabled.

Data stored on a SnapLock storage pool is stored on N series disk devices. In the case of a disaster that destroys the primary filer, you lose access to all the retention-managed data. You can use the SSAM backup storage pool command to back up primary SnapLock-protected storage pools to secondary storage devices, such as tape, either WORM or normal tape. Additionally, you can use the Tivoli Storage Manager Disaster Recovery Manager (Tivoli Storage Manager DRM) feature to move tape volumes off-site for disaster recovery purposes.

Archived



Content Management and integrated Storage Management

In this chapter we discuss how information management products such as Content Manager, Content Manager OnDemand, CommonStore, and Records Manager interact with the underlying storage management layer. We describe the recommended settings and configurations for these integrations:

- ▶ Content Management and Records Management retention requirements
- ▶ Interaction with IBM Tivoli Storage Manager, SSAM, and DR550

9.1 Content and storage management product interactions

IBM content management offerings comprise a variety of products that address diverse aspects of content and information management. Each product offering is specialized and optimized to perform specific functions, as required by the application that utilizes the content management services. Because of this, there are multiple ways in which the IBM content management offerings interact with Tivoli Storage Manager. We discuss how the following products (Figure 9-1) interact with Tivoli Storage Manager:

- ▶ Content Manager
- ▶ Content Manager OnDemand
- ▶ CommonStore
- ▶ Content Manager integrated with Records Manager

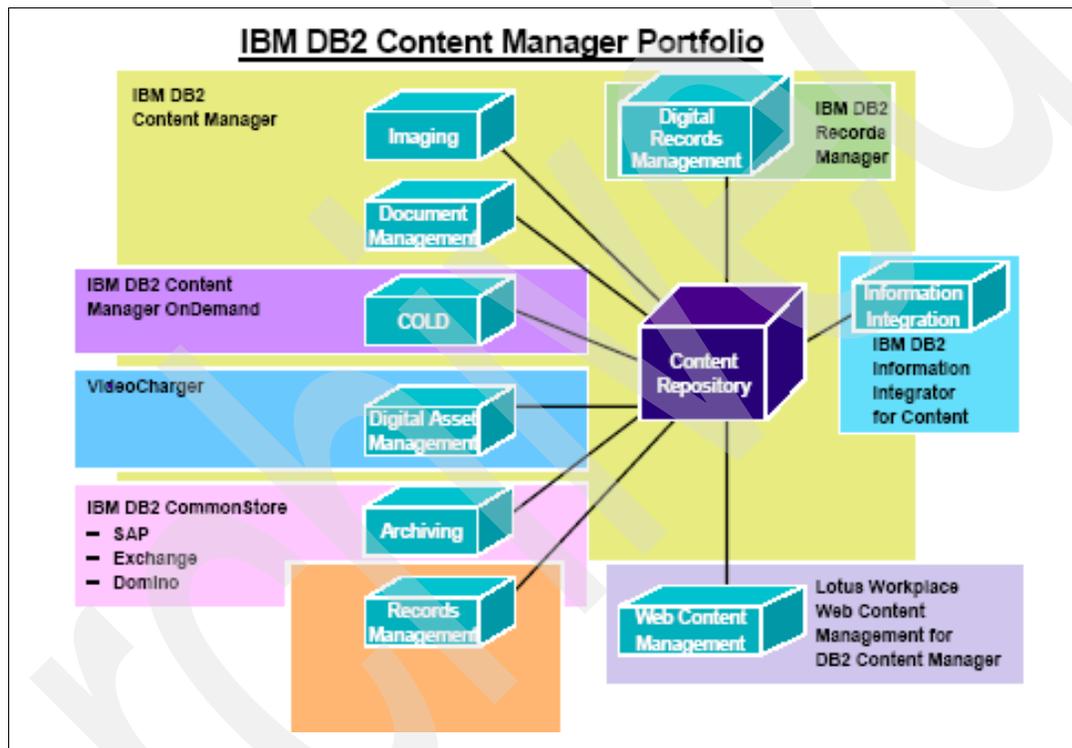


Figure 9-1 Content Repository

We also illustrate some common uses of CommonStore with e-mail and SAP environments, and discuss the differences in using a normal Tivoli Storage Manager server and the SSAM or DR550 solutions.

We consider the following scenario: both the Tivoli Storage Manager (or storage management application) and the content manager application can control the expiration of data stored in the storage management application. The storage management application holds the actual data, whereas the content management application holds the metadata: indexes, descriptions, and pointers to the data.

The content management metadata is used to access the data stored in the storage management application. If the metadata is not available in the content manager application, you will not be able to find the stored data in the storage management application, or the data might be found but might not be usable without metadata information. Therefore, there is a requirement to synchronize the storage management data with the content management metadata.

With a standard Tivoli Storage Manager server, the content manager application has complete control of retention and expiration of objects stored in Tivoli Storage Manager. With SSAM and the DR550, data is stored with a specified retention, and the retention cannot be shortened. In this case you must align the storage management expiration with the content management application expiration, except that you use event-based retention.

Figure 9-2 illustrates the possible interactions of applications such as SAP and e-mail programs with CommonStore and Content Manager and then standard Tivoli Storage Manager or SSAM and the DR550.

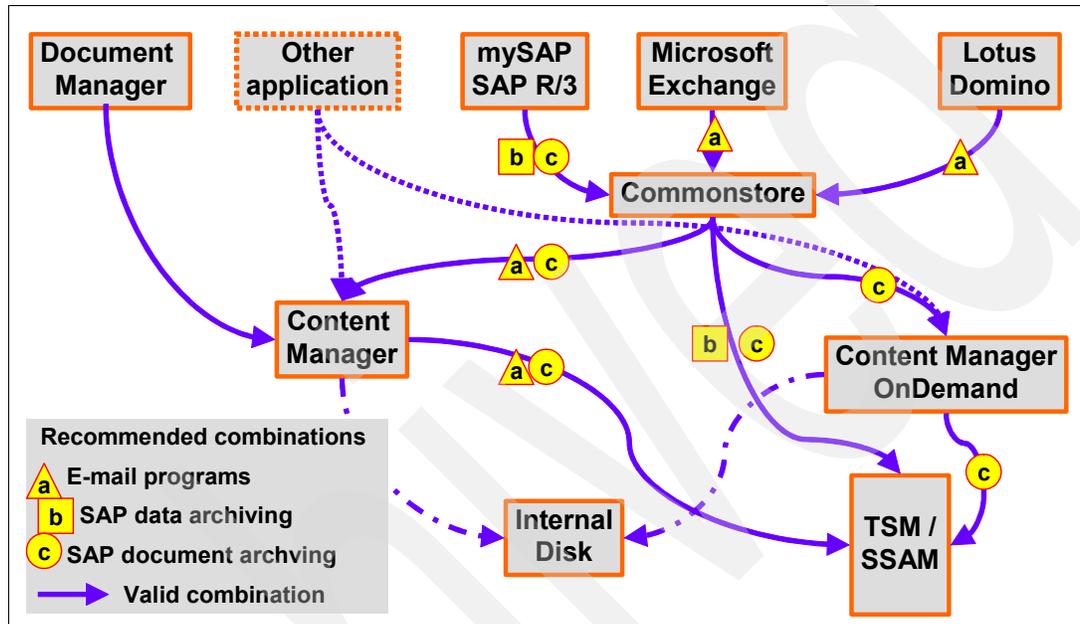


Figure 9-2 Content management and storage management product interactions

The combinations shown in Figure 9-2 are all valid (blue lines). Some combinations are more recommended (yellow signs) because they fit better for the most common customer requirements. Other combinations could be an option as well for different requirements. We discuss these requirements and options later in 9.4, “DB2 CommonStore” on page 236.

As an example, e-mail applications such as Lotus Domino and Microsoft Exchange often have archiving requirements. CommonStore for Lotus Domino (CSLD) and CommonStore for Exchange (CSX) allow you to extract, archive, and delete data such as e-mails or e-mail attachments from the e-mail application’s database. The extracted data can be stored into content repositories such as Tivoli Storage Manager or SSAM or stored into Content Manager, and Content Manager can then archive this data into Tivoli Storage Manager or SSAM.

CommonStore extracts the e-mails and stores them into a content repository as described. Content Manager provides additional functionality such as indexing and searching, which are not provided by Tivoli Storage Manager or SSAM. The recommended approach is to use CommonStore in conjunction with Content Manager because some functionality such as full-text search or single instance store is provided only with this combination.

Therefore, in Figure 9-2 we illustrate e-mail data coming from e-mail applications (a) and flowing to CommonStore, then to Content Manager, and Content Manager will store the data in Tivoli Storage Manager or SSAM. If for example full-text search or single instance store is not required by the customer then Content Manager OnDemand would be an option for the content repository as well.

A second example could be SAP data archiving, where old data of the SAP database is archived and deleted to reduce the growth of the database and to increase the performance of the SAP system. The data is extracted from the SAP database by using an SAP ADK archiving program. CommonStore can pass the data to the content repository. An SAP ADK deletion program removes the archived data from the database afterwards. The extracted data is based on a non-readable form and will be archived for restoring it later if required. It does not make sense to store this data into Content Manager (or Content Manager OnDemand) because it cannot be displayed. Therefore, we recommend that you archive this SAP data into Tivoli Storage Manager or SSAM directly as shown by (b) in Figure 9-2 on page 219.

There are other document types in SAP that could be archived as well, such as incoming business documents (scanned documents) which are part of the business workflow in SAP or printlists. Business documents can be archived either with early archiving or late archiving. With early archiving, documents are scanned first and processed electronically within the workflow. With late archiving, documents are processed as paper within the workflow, scanned later into the system, and linked to the transaction with the help of barcodes. These business documents and printlists can be displayed by a viewer, therefore, it depends on the business requirements which content repository you choose.

The data that is managed by the various IBM content management products is written to Tivoli Storage Manager or SSAM in most cases.

Because all regulations require the protection of the data, there is an implicit requirement to provide a substantively equivalent “non-rewritable, non-erasable, non-alterable” management environment — whether by the content and/or records management application or by the storage management software, storage hardware, or both. Choosing the right components to provide the equivalent “non-rewritable, non-erasable, non-alterable” management environment depends on your business requirements. Therefore, if the requirement is to have a non-erasable, non-rewritable environment on the storage management software level, then you must use SSAM or DR550 (Figure 9-3); alternatively, a standard Tivoli Storage Manager server will suffice.

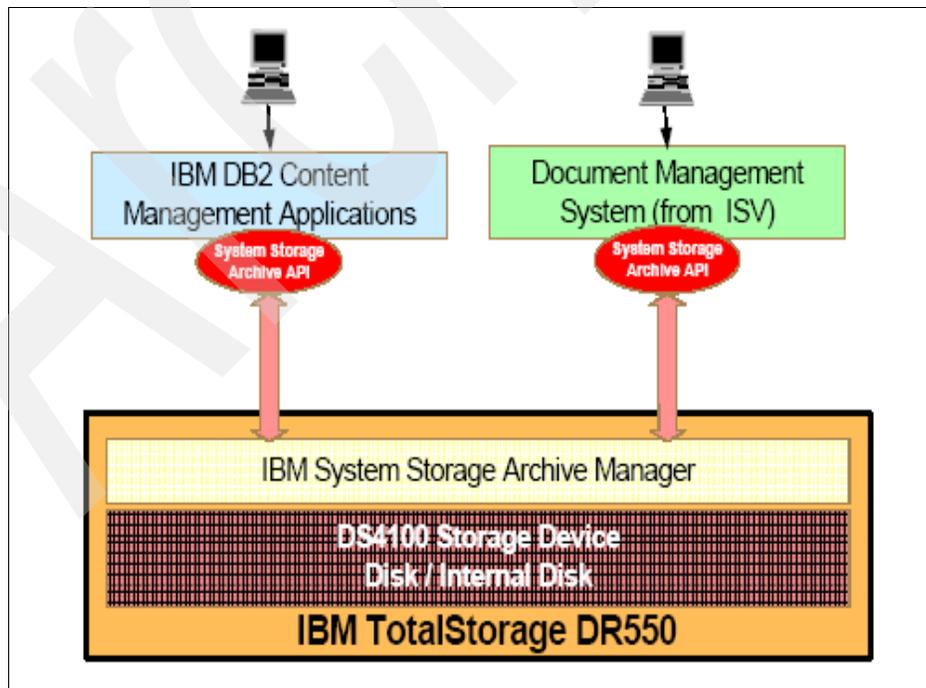


Figure 9-3 DR550

SSAM and the DR550 offer a non-erasable, non-rewriteable storage device, also termed WORM storage, as discussed in Chapter 4, “IBM Tivoli Storage Manager and IBM System Storage Archive Manager” on page 73. WORM storage such as the DR550 functionally allow you to manage retention and expiration in two ways:

- ▶ Chronological retention: You declare a retention period when an object is stored
- ▶ Event based retention: You store the object without specifying a retention, and the object is kept forever or until you signal an event to start counting towards expiration

We recommend that you use event-based retention if possible to avoid independent deletion of data.

Synchronizing content metadata with storage data

It is important to synchronize retention in the content management application and retention in the storage management application, such as SSAM and DR550. Next, we discuss the interaction between content management application metadata and the storage management application data.

The content management application has metadata that allows it to locate and access the data and perform other data management operations such as indexing and search. The content management application sends the data to the storage management application such as SSAM or DR550 and these store and protect the data from tampering and accidental deletion. When you perform a search, the content management application uses the metadata to locate a set of matching data that can then be retrieved from the storage management application. If the metadata is not available or has been deleted, you are not able to perform a search operation and therefore, the data cannot be found. The data might still be present in the storage management application. You could perform a low level scan of the storage management application searching for the records of interest, and, depending on the record type, they might be usable.

SSAM manages retention in a manner that closely emulates a physical WORM storage device. Each data object stored has a retention associated with it, either the chronological *keep until* date type or the event type where retention is not known when the object is stored, but enabled on a later date when a specific event occurs. Also, the retention for an object cannot be shortened, only extended if required. This is at the core of the retention management paradigm and prevents tampering with the archived data, because nobody can intentionally tamper with the record. You ask the DR550 to store data for a set period of time, and DR550 will not allow you to delete the data before expiration.

On the other hand, the content management application, by its very nature, manages the data. The content management application might initially store the data object with a certain retention and at a later time decide to shorten the retention or delete the object before the initial retention period has expired. This operation is allowed in a standard Tivoli Storage Manager server but is not possible in a retention protected SSAM or DR550. If you have asked DR550 to store the object for a year, you are not allowed to delete it before it reaches its expiration. This is true for SSAM chronological retention.

It is possible to configure the content management application for event based retention as well. The content management application stores the object with an undefined retention time and then at a later time it sends an event to delete the object. This is a perfectly acceptable configuration, but you must realize that retention is no longer controlled by the SSAM or DR550 but entirely delegated to the content management application, that can decide to expire the object at any moment. In this case retention is no longer controlled by the hardware device but delegated to the software application. The hardware device will enforce retention by avoiding accidental or intentional deletion but will accept a request from the content management software to delete the object.

Table 9-1 summarizes the various possible product interactions. The type of interaction depends on the ability of the content management product to support event based retention and also on the choice of a normal Tivoli Storage Manager server or a retention protected SSAM or DR550.

Table 9-1 Content and storage management interactions

	Standard IBM Tivoli Storage Manager	SSAM	TSM retention types: E - event based C - chronological
Content Manager	OK, with backup copy group	OK, with archive copy group (from CM 8.2 FP3 and later)	E (only with SSAM)
Content Manager with Records Manager	OK, with backup copy group	OK, with archive copy group (from CM 8.2 FP3 and later). Special considerations apply.	E (only with SSAM)
Content Manager OnDemand	OK, with archive copy group	OK, with archive copy group	E/C
CommonStore	OK, with archive copy group	OK, with archive copy group	C
CommonStore with Content Manager	OK, as with Content Manager	OK, as with Content Manager	E (only with SSAM)
CommonStore with Content Manager OnDemand	OK, as with Content Manager OnDemand	OK, as with Content Manager OnDemand	E/C

9.2 DB2 Content Manager, Tivoli Storage Manager, and SSAM

Until Content Manager Version 8.2.3, Content Manager has supported storing data into Tivoli Storage Manager using the Tivoli Storage Manager backup API interface only. Therefore, it could use backup copy groups in Tivoli Storage Manager only. Backup copy groups do not have automated expiration processing in Tivoli Storage Manager. The data in a backup copy group only expires when an application (such as Content Manager) issues a Tivoli Storage Manager API call to delete the data. In addition, only one Tivoli Storage Manager server was supported within one Content Manager system.

With Tivoli Storage Manager Version 5.2.2, a new kind of Tivoli Storage Manager was introduced, formerly called IBM Tivoli Storage Manager for Data Retention, which is now rebranded to IBM System Storage Archive Manager (SSAM). This version of Tivoli Storage Manager has Data Retention Protection enabled. This ensures that objects that have been archived are not deleted from the Tivoli Storage Manager server until the retention policies set for that object have been satisfied. SSAM actively inhibits deletion of unexpired objects. SSAM only supports the archive API. For more details about SSAM, see Chapter 4, “IBM Tivoli Storage Manager and IBM System Storage Archive Manager” on page 73.

Note: IBM Tivoli Storage Manager support is bundled for free with Content Manager. But the licence for SSAM is not included.

To support SSAM (and solutions such as IBM DRx50) Content Manager introduced the support of the archive API with Content Manager Version 8.2.3. Also a new setup option (in the administration client) was added for defining multiple Tivoli Storage Manager servers within one Content Manager system.

Restriction: Content Manager supports the Tivoli Storage Manager archive API only within an SSAM server, not within a Tivoli Storage Manager server. Also, it supports only the event-based retention mode of SSAM and not the chronological retention mode.

A single Content Manager Resource Manager can now manage Content Manager volumes from two different Tivoli Storage Manager servers. This function allows the customer to have Tivoli Storage Manager volumes with and without retention protection on the same Resource Manager. The access to the Tivoli Storage Manager has not been changed. The Content Manager still uses the backup API to store objects to Tivoli Storage Manager. But it can store objects now to SSAM using the archive API.

Figure 9-4 illustrates the integration of Content Manager and Tivoli Storage Manager / SSAM.

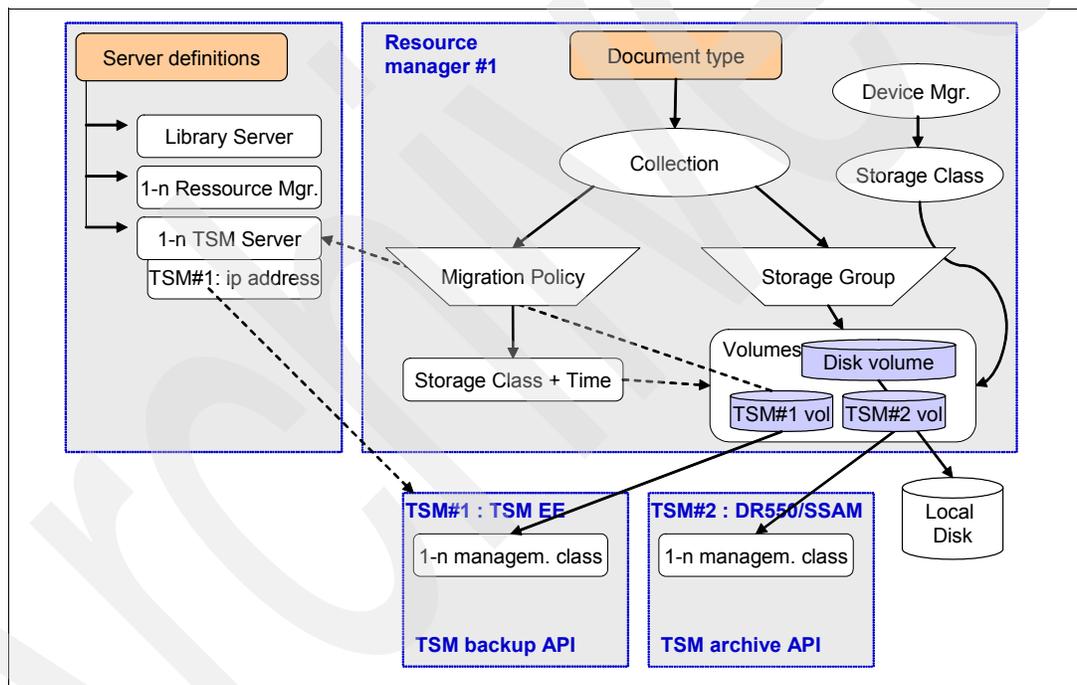


Figure 9-4 Content Manager and Tivoli Storage Manager integration

Content Manager has its own configuration for object storage, called System Managed Storage (SMS). This configuration consists of the following entities:

- ▶ **Device Manager:** A device manager is used by the resource manager to communicate with the actual physical storage location of documents and is comparable to the SCSI and IDE device drivers of the operating system. There are preinstalled device managers such as ICMADDM (with class “TSM”) for accessing Tivoli Storage Manager server.
- ▶ **Storage Class:** A storage class identifies the destination and type of media that an object is stored on. Storage classes can be associated with either a local or remote destination. A local destination is specified by associating the storage class with a device manager. A remote destination is specified by associating the storage class with remote resource manager.

- ▶ **Storage Systems:** A storage system specifies the location, or volume, of where an object is stored, and is directly associated with a storage class. Therefore, in order to define a new storage system volume, you must first define the associated storage class. There are different types of storage systems such as file system volumes or Tivoli Storage Manager volumes.
- ▶ **Storage Groups:** Storage groups are used to specify which storage system volumes can be used by the resource manager to store documents. Storage groups can contain more than one storage system volume.
- ▶ **Migration Policies:** Migration policies specifies the rules for migrating documents between storage classes. They consist of a series of steps that a document will take, and specify how long a document will remain at each storage location.
- ▶ **Collections:** A collection consists of a storage group and a migration policy, and is the object storage entity specified for storing new documents into the resource manager. The location for documents in a collection are derived from the storage group and migration policy. Recall that a storage group defines which storage system volumes a particular document can be stored on, and that the migration policy defines the set of rules for moving the document between storage classes. A collection is used to store similar documents in a common location.

The following rules apply to these Content Manager definitions when using SSAM:

- ▶ You cannot migrate data out of Content Manager volumes that are Tivoli Storage Manager volumes under retention control (SSAM).
- ▶ You cannot have more than one local Content Manager storage class in a Content Manager policy where the primary storage class contains an SSAM volume.
- ▶ If the first Content Manager storage class in the Content Manager policy does not have an SSAM volume, you can:
 - Have other storage classes. In that case, if you also have a storage class with an SSAM volume, it must be the last storage class.
 - Have a remote storage class that contains an SSAM volume.
- ▶ Because Version 8.3.1 Content Manager offers a feature for object aggregation to improve performance with SSAM volumes, this feature is not available when using Tivoli Storage Manager with Content Manager.

There are no restrictions on Content Manager replication, because the source or target collections can have migration policies with an SSAM volume.

To configure Content Manager to work with Tivoli Storage Manager or SSAM, you must have:

- ▶ A Tivoli Storage Manager / SSAM server installed and configured with policy domains, policy sets, management classes, and so on
- ▶ A node registered in that Tivoli Storage Manager / SSAM policy domain
- ▶ The Tivoli Storage Manager client API software (Version 5.2.2 or later) be installed and configured on the Content Manager Resource Manager server

There are two points within Content Manager administration where Content Manager definitions are linked directly to the Tivoli Storage Manager / SSAM server, as illustrated in Figure 9-5.

First, Content Manager has to know the Tivoli Storage Manager / SSAM server (including IP address, node name, password of this node). The Content Manager Resource Manager uses the Tivoli Storage Manager client API software and the definitions in the file “dsm.opt” (Example 9-1 to access the Tivoli Storage Manager / SSAM server).

Example 9-1 dsm.opt file

```
SERvername server_a
  COMMmethod TCPip
  TCPport 1500
  TCPserveraddress SVLTSM1.SVL.IBM.COM
nodename itso.stl.ibm.com
tcpwindowsize 63
```

Later, when setting up system managed storage entities for Content Manager, you create a Tivoli Storage Manager volume inside of Content Manager. This does not create a volume or management class in Tivoli Storage Manager or SSAM. It is a link only. The management class in Tivoli Storage Manager / SSAM has to be defined before. The name (first field) has to be the same name as the appropriate management class in Tivoli Storage Manager or SSAM server. The Content Manager Resource Manager checks at the time of creation (of this link) if this management class is defined in Tivoli Storage Manager or SSAM. It will return an error if not.

Important: Always type your Tivoli Storage Manager management class in uppercase.

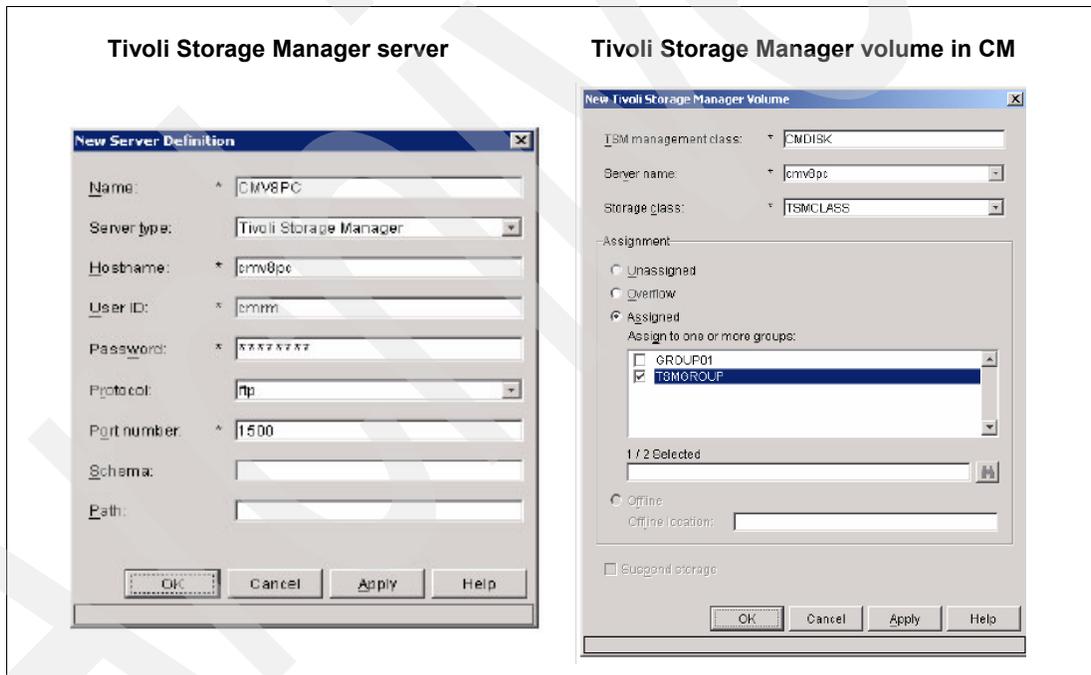


Figure 9-5 Content Manager links to Tivoli Storage Manager / SSAM

Therefore, how does Content Manager now distinguish between the access to Tivoli Storage Manager and SSAM? Remember that Content Manager uses the Tivoli Storage Manager backup API to access Tivoli Storage Manager, and it uses the Tivoli Storage Manager archive API to access SSAM (only with event-based retention).

The definition of the correct access method is located within the Content Manager device manager entity. Figure 9-6 shows examples for both access methods.

Both definitions include Class="TSM". The field "Parameter" has to be empty when using Tivoli Storage Manager server and has to be set to "mode=retention" when using an SSAM server.

Tip: By using this parameter, you do not have to configure the Tivoli Storage Manager API options file with: ENABLEARCHIVERETENTIONPROTECTION ON.

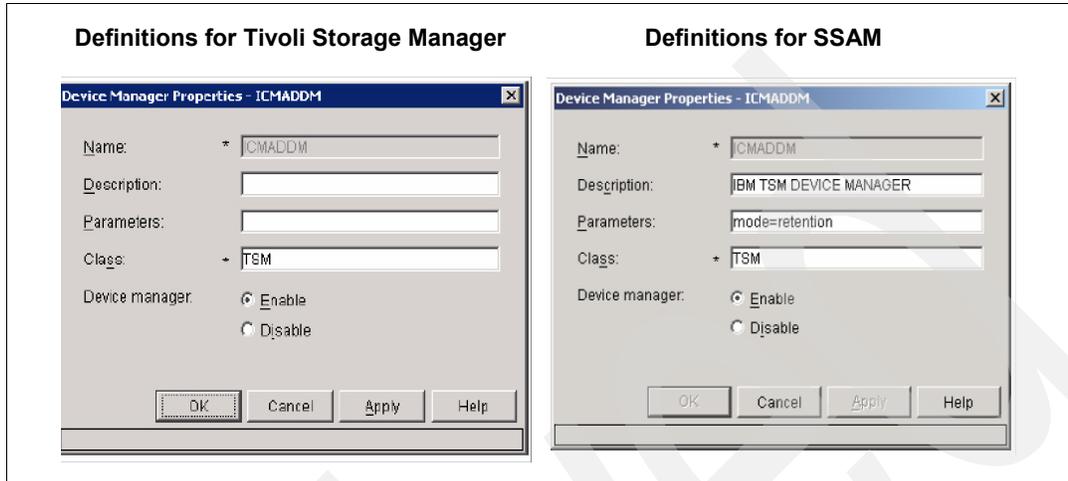


Figure 9-6 Content Manager Device Manager for Tivoli Storage Manager and SSAM

For more details on how to set up Content Manager with SSAM, see Section 5.3 in the IBM Redbook, *Understanding the IBM TotalStorage DR550*, or refer to the *Content Manager System Administration Certification Study Guide*.

Now that we understand how Content Manager accesses Tivoli Storage Manager and SSAM server — how is the retention of objects managed, and how are documents deleted?

Let us start with the delete process. When using Tivoli Storage Manager with Content Manager, backup copy groups will be used only. Backup copy groups do not have automated expiration processing in Tivoli Storage Manager. The data in a backup copy group will only expire when an application (such as Content Manager) issues a Tivoli Storage Manager API call to delete the data. Therefore, when a Content Manager user or administrator requests to delete documents (we assume that he has the proper access rights to do so) a Tivoli Storage Manager API call will be issued in the follow-on process inside of Content Manager Resource Manager (one job of the migrator process) to delete the appropriate data.

A new delete method was introduced with the support of the Tivoli Storage Manager archive API. Content Manager supports only the event-based retention mode in conjunction with the Tivoli Storage Manager archive API.

Figure 9-7 shows a timeline depicting event-based policy. In this example, Content Manager archives data using the retention values RETINIT=Event, RETMIN=360 and RETVER=0. We recommend that you set RETMIN to 0 if compliance is not required, or to the minimum supposed retention period in environments where compliance is required. The parameters RETMIN, RETINIT, and RETVER will be set in the configuration of the appropriate SSAM or Tivoli Storage Manager management class.

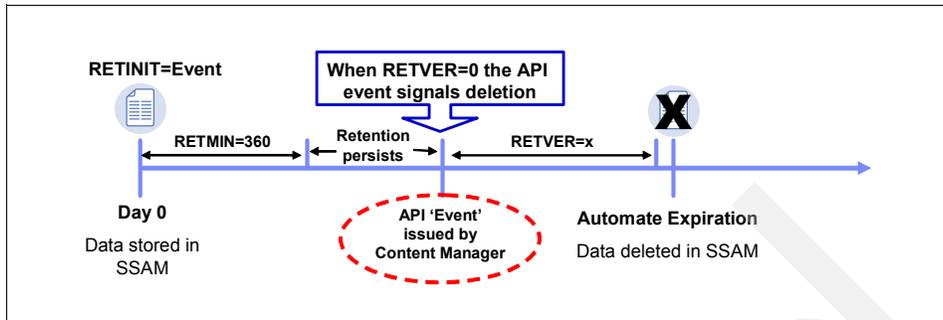


Figure 9-7 Tivoli Storage Manager event-based retention

Content Manager sends the API “event” call to delete the object and not to activate the retention period. The data is retained a minimum of 360 days (RETMIN) in this example and gets deleted immediately after the API “event” call if RETMIN was expired. This makes sense because the Content Manager metadata was deleted already at this point of time. It is possible to set RETVER bigger than 0 so that authorized users or administrators still have the possibility to access the data over a defined transition period.

In the case that the API “event” call was issued before RETMIN was expired SSAM will store the receipt of the API “event” call and will delete the data after RETMIN expired based on the example values explained previously.

Restriction: Content Manager does not support deletion hold and release feature of SSAM.

Another important topic is the management of WORM storage when deleting objects. Companies have to dispose and destroy documents after their legal and business uses are complete (retention time). Content Manager V8.1 introduced the ability to assign retention periods to Content Manager item types. Item types are used as container for defining and categorizing documents (items) created in Content Manager. Administrators can configure retention times by specifying the proper retention period when an item type is defined within Content Manager. The retention period should be consistent with the customer’s retention policies for the type of documents associated with the specific item type. An example is: “All invoices received should be retained for seven years from the date they are received.”

When a document of that Content Manager item type is created, the library server will automatically calculate the expiration date. This is stored as a system attribute (date field). Content Manager does not delete expired items automatically. In order to expire data in Content Manager you would create a small application that has not much more than 50 lines of JAVA. Authorized users could use such an application to search for the expired documents, generate reports (to get management approval, for example) and subsequently delete the expired items. Content Manager would then delete the index rows in their database for these documents and tell Tivoli Storage Manager / SSAM to delete these documents. Because of this, Content Manager as the application controls the retention of objects and not the storage management layer, such as Tivoli Storage Manager or SSAM.

It is possible to define a minimum retention in SSAM server in addition (if compliance is required). The retention definition stored in Content Manager item types will not be synchronized with the retmin value in Tivoli Storage Manager / SSAM. You have to set up the Content Manager item types and management classes in SSAM in the correct manner to store data of specific item types to the appropriate management classes in SSAM.

Important: Without IBM Records Manager or a similar developed application, there is no ability to create a legal hold, suspend, wait on an event, or keep detailed tracking of user access, modifications, and deletions. Development of such an application is not a trivial process and is not recommended. This solution does not have a records management component that is certified by any authority.

With the introduction of Records Manager and its integration with Content Manager, Records Manager became an alternative for controlling the disposition of data within Content Manager. We discuss this later in 9.5.2, “DB2 CM and Storage Management together with DB2 Records Manager” on page 251.

DB2 Content Manager z/OS

Until Version 8.3, Content Manager z/OS Version used the Object Access Method (OAM) to store data. OAM manages large objects, such as compressed scanned images or coded data, in their entirety and contains no restrictions on the data in an object. Objects might be stored on disk, tape, or optical platters, and freely staged up and down the storage hierarchy based on system managed storage (SMS) parameters.

The support for Tivoli Storage Manager was introduced with Content Manager z/OS Version 8.3. This integration is using the Tivoli Storage Manager API client V5.2.2 (or greater) to connect to a Tivoli Storage Manager server. The FP1 level of PTF of Content Manager z/OS Version 8.3 is the minimum level required for this integration. It is now possible to use Tivoli Storage Manager instead of OAM or in addition to OAM in order to take advantage of Tivoli Storage Manager facilities (different collections from the Library Server perspective) as shown in Figure 9-8.

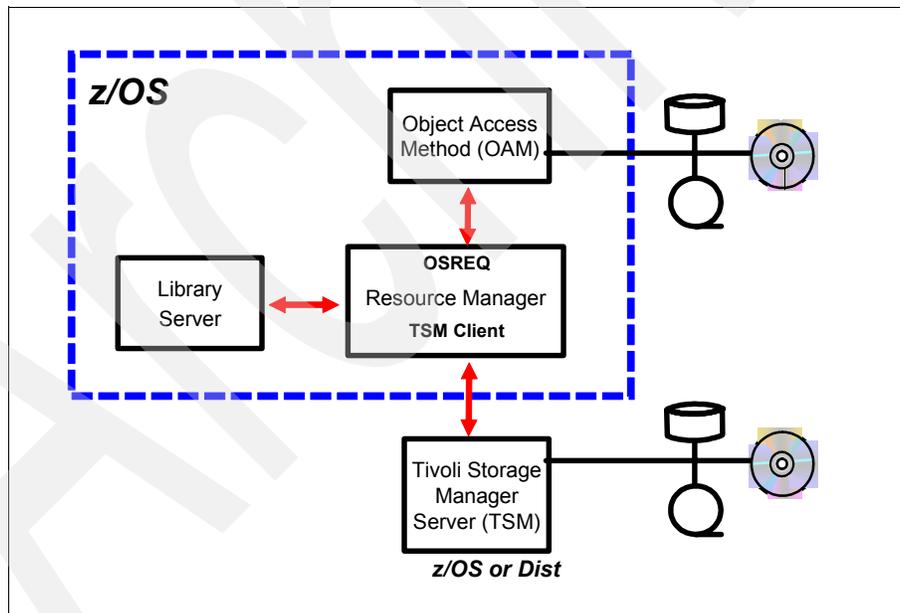


Figure 9-8 TSM/OAM Support

Only one Tivoli Storage Manager server instance (z/OS or Distributed) can be defined to one Resource Manager. In order to enable Content Manager z/OS to use Tivoli Storage Manager, a Tivoli Storage Manager server must be installed and configured to allow Content Manager z/OS to store and retrieve objects. In addition, the Tivoli Storage Manager OS/390® UNIX System Services Client API Version 5.2.2 or greater must be installed on the Content Manager z/OS server.

For more details on how to set up Content Manager z/OS with Tivoli Storage Manager or SSAM, see *Content Manager z/OS V8.3 Installation Guide*, GC18-7698-02. Content Manager on z/OS supports the backup and archive API of Tivoli Storage Manager, unlike Content Manager on Distributed platforms.

Restriction: Aggregation of objects with Content Manager on Distributed platforms is not supported with Content Manager z/OS.

Figure 9-9 shows a possible configuration of one Library Server with several Resource Managers connected to different Tivoli Storage Manager server within one Content Manager system. Normally one z/OS Tivoli Storage Manager Server is defined per LPAR.

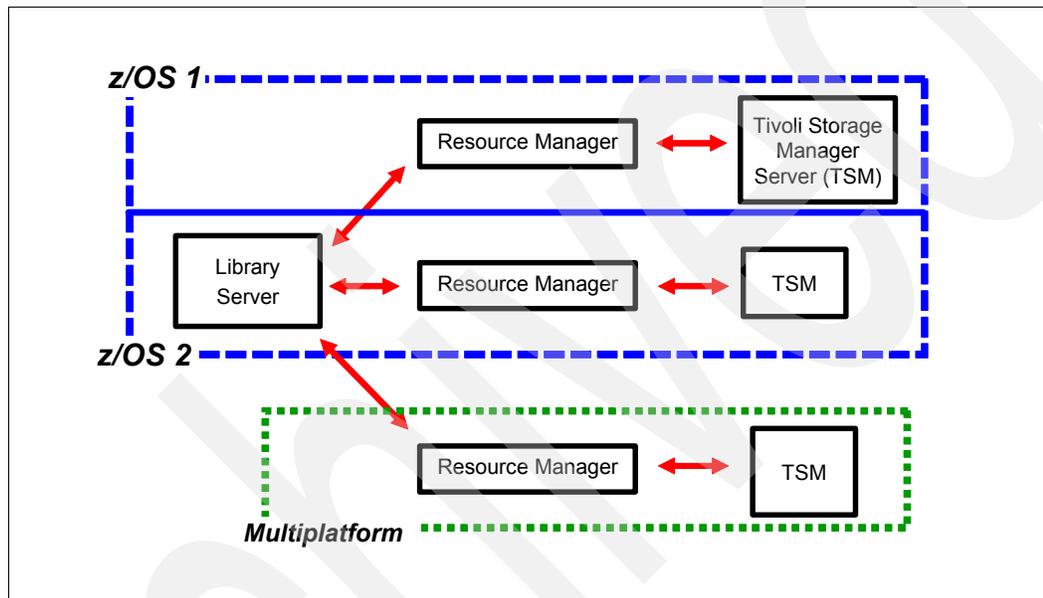


Figure 9-9 Resource Manager and TSM

After introducing the support of Tivoli Storage Manager, how can objects be moved between OAM and TSM? This is possible with programming. The z/OS RM accepts a **changeSMS** API call that allows the user to change the collection for an object. The collection can be changed, for example, from an OAM collection to a TSM collection. The result is that the object is copied to the TSM collection and deleted from the OAM collection. For more details, see *Content Manager V8.3 Application Programming Guide*, SC27-1347-04.

9.3 DB2 Content Manager OnDemand

Content Manager OnDemand consists of a library server and one or more object server. The system components which are required for creating, retrieving, and viewing an OnDemand report are an application, an application group, a storage set and a folder. These elements, in combination, allow the OnDemand administrator to define and create a report definition which can then be used to index and load data into OnDemand.

Application: An application describes the physical characteristics of a report to OnDemand. Typically you define an application for each program that produces output to be stored in OnDemand. The application includes information about the format of the data, the orientation of data on the page, the paper size, the record length, and the code page of the data. The application also includes parameters that the indexing program uses to locate and extract

index data and processing instructions that OnDemand uses to load index data in the database and documents on storage volumes.

Application groups: An application group is a collection of one or more applications which contain common indexing and storage management requirements. The application group contains the database information which is used to load, search for, and retrieve reports. The application group defines the data which to be loaded into the database. In the following sections we take a closer look at aspects of application group definition which can contribute to a successful OnDemand system implementation.

Storage sets: A storage set contains one or more storage nodes that can be used by several application groups which have the same archive storage requirements. For example, a storage set can be used to maintain data from different application groups that have to retain documents for the same length of time and require the data to be kept on the same type of media. Different storage sets can be created to handle different data retention requirements. One storage set could be set up to maintain data on cache-only storage, another could be set up to point to an archive storage to maintain data for three years on optical media. Business practices and legal requirements determine the storage management design required. Content Manager OnDemand supports Tivoli Storage Manager as their archive repository and use the Tivoli Storage Manager archive API to communicate with and transfer data objects to archive storage.

Folder: A folder is the user's way to query and retrieve data stored in OnDemand. A folder provides users with a convenient way to find related information stored in OnDemand, regardless of the source of the information or how the data was prepared. A folder allows an administrator to set up a common query screen for several application groups that might use different indexing schemes, so that a user can retrieve the data with a single query. For example, a folder called Student Information might contain transcripts, bills, and grades, which represents information stored in different application groups, defined in different applications, and created by different programs.

In the storage management definition of the OnDemand library server you can specify where and when OnDemand stores reports and how those reports are maintained. Figure 9-10 illustrates OnDemand storage object relationships. When a report is loaded into OnDemand, it is assigned to an application group. The application group is associated with a storage set. The storage set contains one or more storage nodes that can be used by several application groups which have the same archive storage requirements.

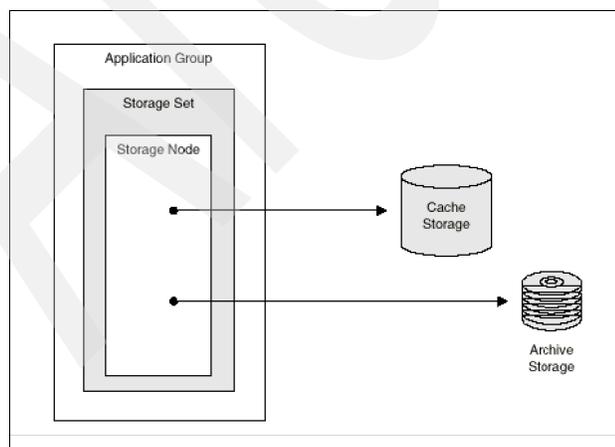


Figure 9-10 Content Manager OnDemand storage objects

For example, a storage set can be used to maintain data from different application groups that have to retain documents for the same length of time and require the data to be kept on the same type of media. Different storage sets can be created to handle different data retention requirements. One storage set can be set up to maintain data on cache only Direct Access storage. Another can be set up to point to a Tivoli Storage Manager client node that will cause a copy of the report to be stored in archive storage.

If Tivoli Storage Manager is being used as the archive storage manager, the same storage management criteria should be specified for both OnDemand and Tivoli Storage Manager. That is, the Life of Data and Indexes in OnDemand and the retention period in Tivoli Storage Manager should be the same value.

Note: The date that is used to determine the Life of Data and Indexes in OnDemand is the date field index value taken from the report being loaded. The date used for the retention period in Tivoli Storage Manager is the date that the report is first migrated to Tivoli Storage Manager. If the load type value for the application group is Load, a command is issued from OnDemand to Tivoli Storage Manager to delete data when the data is being expired from OnDemand. If the load type is segment or document, a delete command is not issued from OnDemand to Tivoli Storage Manager when OnDemand expires the data and the data remains in Tivoli Storage Manager until the Tivoli Storage Manager retention period expires. This data will not be accessible from OnDemand due to the fact that the indexes have been expired in OnDemand.

Storage set definition

A storage set can contain one or more primary storage nodes. A primary storage node is used to manage reports and resources stored in an application group. A storage node is associated with a specific OnDemand object server. When Tivoli Storage Manager is used for archive storage, each storage node associated with Tivoli Storage Manager managed storage must be registered as a client node in a Tivoli Storage Manager policy domain. The Tivoli Storage Manager policy domain properties determine the type of storage devices that are used to maintain the archived data and the length of time that the data is maintained.

OnDemand systems can be set up to run as cache only Direct Access storage systems with no migration of the data or indexes, or with an archive system utilizing Tivoli Storage Manager to maintain and manager the archive of OnDemand documents and indexes over a pre-designated period of time.

When OnDemand is installed and the system is initialized, a default cache only storage set is created. Additional cache storage sets can be defined. Storage sets associated with Tivoli Storage Manager client nodes that are tied to specific management policies on the Tivoli Storage Manager servers are used for long term archive storage. The OnDemand administrator defines and maintains storage sets (Figure 9-11). The load type is the storage set parameter that we examine here.

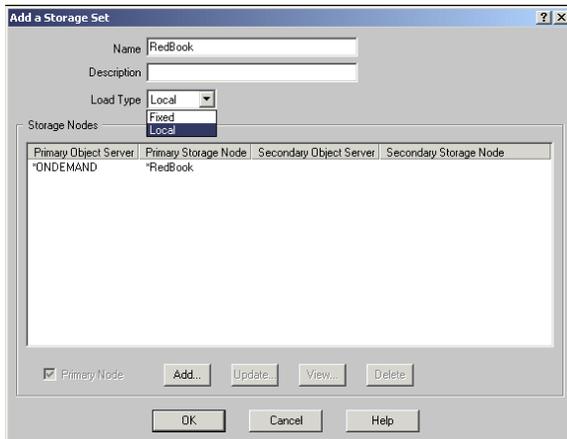


Figure 9-11 Storage set definition

The load type parameter determines where OnDemand stores data. There are two possible values:

- ▶ **Fixed:** OnDemand stores data in the primary storage node that has the load data field selected. When load type is set to fixed, you must select the load data check box for one primary storage node. OnDemand loads data to only one primary storage node regardless of the number of primary nodes that are defined in the storage set.
- ▶ **Local:** OnDemand stores data in a primary storage node on the server on which the data loading program executes. When load type is local, the load data check box must be selected for a primary storage node on each of the object servers which is identified in the storage set. A storage set can contain one or more primary storage nodes that reside on one or more object servers.

On the primary node panel (Figure 5-11), there are several parameters that we have to examine.

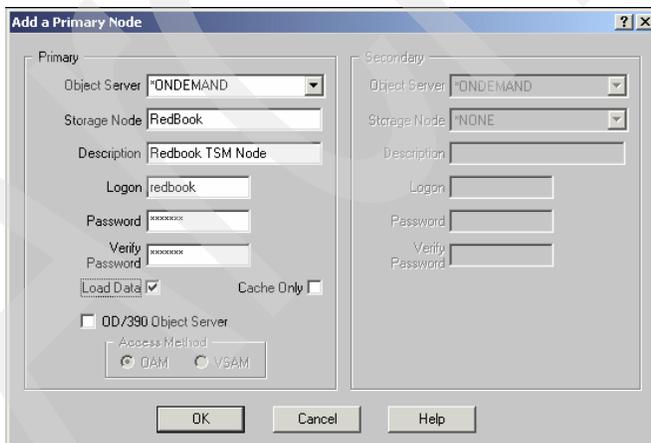


Figure 9-12 Primary node definition

Note: The OnDemand storage node name does not tie the storage set to the Tivoli Storage Manager client node. This name is only a label in the OnDemand system. The storage node name can be the same as the associated client node name, but it is not required that they be the same.

If Tivoli Storage Manager is being used to maintain archive data, the logon parameter is the name of the Tivoli Storage Manager client node. This parameter is ignored if you are defining a cache only storage node. The logon field must be a valid Tivoli Storage Manager client node name. The password which follows the logon must be the same as the password created for the client node. OnDemand uses a Tivoli Storage Manager archive API to connect and logon to the Tivoli Storage Manager server when data is being migrated to the Tivoli Storage Manager client node.

The load data parameter determines the primary storage node into which OnDemand loads data. When the load type is fixed, one primary storage node must have load data selected. When load type is local, load data must be selected for one primary node for each object server that is associated with the storage set.

The cache only parameter determines whether OnDemand uses the archive manager for long term storage of data. After installing and configuring Tivoli Storage Manager, creating an OnDemand storage set, and assigning it to a Tivoli Storage Manager client node, we are ready to consider how an application group uses the cache storage manager and the archive storage manager to store, maintain, and expire OnDemand report data.

Application group storage management

The application group storage management settings (Figure 9-13) determine how long report data and indexes are kept in cache storage before being expired. There are also choices to be made concerning how soon data is migrated to the archive storage after the report load is completed.

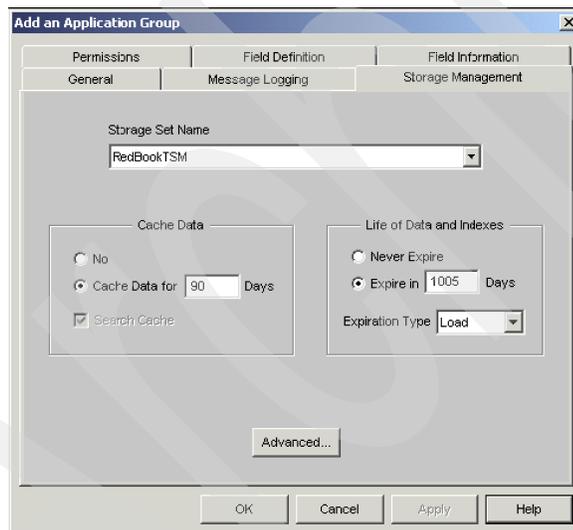


Figure 9-13 Application group storage management

The cache data setting determines if the report data is stored in Disk cache and, if so, how long it is kept in the cache before it expires. You can also choose to have cache searched or not searched when retrieving documents for viewing. If you choose not to store reports in cache, a storage set that supports archive storage must be selected.

Note: Data that is retrieved often should generally remain in cache until it is no longer required by 90% of OnDemand users.

The life of data and indexes settings determine the length of time that report data, indexes and resources are maintained in the OnDemand system before they are deleted from the application group. The report data, indexes, and resources can be maintained indefinitely if set to never expire, or might be kept for up to 273 years. After the maintenance threshold has been reached, the `arsmaint` command can be used to expire the data from the system.

The expiration type determines how report data, indexes, and resources are expired. There are three expiration types:

- ▶ **Load:** If the expiration type is load, an input file at a time can be deleted from the application group. The latest date in the input data and the life of data and indexes determines when OnDemand deletes the data. Data that has been stored in archive storage is deleted by the storage manager based on the archive expiration date. Load is the recommended expiration type.
- ▶ **Segment:** If the expiration type is segment, a segment of data at a time is deleted from the application group. The segment must be closed and the expiration date of every record in the segment must have been reached. If small amounts of data are loaded into the application group, and the maximum rows value is high, the segment might be open for a long period of time and the data is not be expired for the period.
- ▶ **Document:** If the expiration type is document, a document at a time is deleted from the application group. Storing with an expiration type of document causes the expiration process to search through every document in the segment to determine if the expiration date has been reached resulting in long processing times.

Retention and disposition

Retention and disposition of data in Content Manager OnDemand depends on the expiration type defined in the application group.

If you specified expiration type to be “segment” or “document” OnDemand will not trigger Tivoli Storage Manager directly to delete objects when deletion or expiration of objects in OnDemand occurs (disconnected process). Content Manager OnDemand stores a life of data value in the index such that when the document reaches its life of data period, information about it is removed from the OnDemand database (the document can no longer be retrieved). Content Manager OnDemand and Tivoli Storage Manager delete documents independently of each other. This is because deletion of type document means individual rows might expire. Because most objects stored contain more than one document, the object can only be deleted when all the documents have expired.

Keeping track of such information would be very complex. Segment deletions are similar. If a segment is expired, it might span dozens or hundreds of stored objects. An object might also span segments. OnDemand would have to scan all the segment tables to see if a particular object was safe to delete. This would be very time intensive and prohibitive in extremely large systems. In this case Content Manager OnDemand and Tivoli Storage Manager use their own criteria to determine when documents expire and use their own utilities to remove documents from the system. The Life of Data parameter, which is used by OnDemand, and the Retention Period, which is used by the Tivoli Storage Manager, are the same value so that documents get deleted from both places at the same time if correct defined.

If you specified expiration type to be “load” OnDemand can trigger Tivoli Storage Manager directly to delete objects when deletion or expiration of objects in OnDemand occurs. If that is the case then Content Manager OnDemand will expire documents in a particular load (when they are eligible to be expired) by deleting the index rows for the documents and by issuing delete commands through the archive API to Tivoli Storage Manager so that Tivoli Storage Manager will delete the objects that contain the documents.

Content Manager OnDemand supports also SSAM server for archiving objects because its integration with Tivoli Storage Manager is based on the Tivoli Storage Manager archive API. In order to work with an SSAM server, the Tivoli Storage Manager client sample file that is shipped with Content Manager OnDemand has to have the following parameter set: ENABLEARCHIVERETENTIONPROTECTION ON. For details on how to set up Content Manager OnDemand with SSAM, see Section 5.4 in the IBM Redbook, *Understanding the IBM TotalStorage DR550*.

Content Manager OnDemand does support the event-based retention model of Tivoli Storage Manager with Content Manager OnDemand version 7.1.2.2 or later. But event-based retention only works with expiration type “load” because Content Manager OnDemand can only trigger Tivoli Storage Manager server directly when using expiration type “load”.

Therefore, Content Manager OnDemand can expire documents in a particular load with this enhancement and can now issue an API “event” call (just as Content Manager does) so that the objects will be removed from a Tivoli Storage Manager server. Reports are deleted from Tivoli Storage Manager when the application group's expiration type is load AND when the “life of data and indexes” expires. Otherwise, if you specified expiration type to be “segment” or “document”, expiration never occurs in Tivoli Storage Manager. Therefore, we do not recommend you to use event-based retention with expiration type “segment” or “document”.

Restriction: Content Manager OnDemand does not support deletion hold and release feature of SSAM.

We recommend that you use expire by load and event-based retention if possible to avoid independent deletion of data.

Note: When integrating CommonStore with Content Manager OnDemand expiration type “load” (and as the result event-based retention model) is not supported.

Content Manager OnDemand as the application (when using expiration type “load”) controls the retention of objects and not the storage management layer such as Tivoli Storage Manager or SSAM. It is possible to define a minimum retention in SSAM server in addition if required. The retention definition stored in Content Manager OnDemand will not be synchronized with the retmin value in Tivoli Storage Manager / SSAM. You have to set up the Content Manager OnDemand application group and management classes in SSAM in the correct manner to store data of specific item types to the appropriate management classes in SSAM. An example definition could be:

Content Manager OnDemand application group - life of data and indexes: 360 days

SSAM definition: RETINIT = EVENT, RETMIN = 360 days and RETVER = 0.

In that example, Tivoli Storage Manager will not expire the loaded data until it is expired or unloaded by OnDemand. If you delete the applications group, unload the data, or let it expire normally, OnDemand sends the event trigger to Tivoli Storage Manager to clean up the appropriate data.

If you use expiration type “segment” or “document” and chronological retention in Tivoli Storage Manager the same storage management criteria should be specified for both OnDemand and Tivoli Storage Manager. That is, the Life of Data and Indexes in OnDemand and the retention period in Tivoli Storage Manager.

ARSMaint

The ARSMaint program maintains application group data that is stored in the OnDemand database and in cache storage. It maintains the system using the storage management values that are specified for application groups. It is typically run in a regular schedule to migrate documents from cache storage to archive storage, migrate index data to archive storage, and delete documents from cache storage and index data from the OnDemand database. arsmaint uses the application group expiration type to determine how to delete index data from an application group. arsmaint can expire a table of application group data at a time (segment expiration type), an input file of data at a time (load expiration type), or individual documents (document expiration type).

Note: When expiring cache data, by default, the data is not expired until the cache storage file system has exceeded 80 percent of capacity. Keeping data in cache as long as possible improves retrieval and viewing performance. You can force the expiration of cache data before cache is 80 percent full by using the minimum and maximum parameters to override the percentage full default. Refer to IBM Content Manager OnDemand for Multiplatforms - Administrator's Guide, SC27-0840 for detailed explanation of the arsmaint command and its associated parameters, along with all other OnDemand commands.

Content Manager OnDemand z/OS

Content Manager OnDemand z/OS used the Object Access Method (OAM/VSAM) to store data. OAM manages large objects, such as compressed scanned images or coded data, in their entirety and contains no restrictions on the data in an object. Objects can be stored on disk, tape, or optical platters, and freely staged up and down the storage hierarchy based on system managed storage (SMS) parameters.

The support for Tivoli Storage Manager was introduced for Content Manager OnDemand z/OS version 7.1 with APAR PQ92029 (included in SPE-4). This integration is using the Tivoli Storage Manager OS/390 Unix System Services API client to connect to a Tivoli Storage Manager server.

In order to enable Content Manager OnDemand z/OS to use Tivoli Storage Manager, a Tivoli Storage Manager server must be installed and configured to allow Content Manager z/OS to store and retrieve objects.

In addition, the Tivoli Storage Manager OS/390 Unix System Services Client API Version 5.2.2 or greater must be installed on the Content Manager OnDemand z/OS server.

To define Content Manager OnDemand storage nodes that use Tivoli Storage Manager, you must have the OnDemand Administrative Client version 7.1.2 or later (US English) installed.

9.4 DB2 CommonStore

The DB2 CommonStore product family supports three different backend archives: Content Manager, Content Manager OnDemand, and Tivoli Storage Manager/SSAM. Each of these archive options has a unique architecture and particular strengths.

Some functional features and options in CommonStore depend on the backend archive option. This chapter provides a detailed explanation of these differences and how they impact security, indexing, workflow, and so on.

Furthermore, some of the technical aspects of the archive system itself such as storage, data compression, and document removal differ in conjunction with the CommonStore solution.

Storage

Both Content Manager and Content Manager OnDemand can manage archived items in a file system on hard disk. For long-term storage, both repositories pass on the archived items to Tivoli Storage Manager. Whatever backend repository you choose, the archived items will finally end up in Tivoli Storage Manager. Therefore, do the three different backend repositories differ at all with respect to storage?

Yes, the three different backend repositories store each archived item in a very particular way. As an example, let's have a look at archiving Outlook e-mails with CommonStore for Exchange. One hundred e-mails, each 80 KB in size, are to be archived and stored in Content Manager, Content Manager OnDemand or Tivoli Storage Manager. The Feature for Single Instance Store is not activated (for better comparison).

In Content Manager, each e-mail is stored as a separate item. This means that there are 100 entries in the Content Manager library and 100 items in the Content Manager file system on the resource manager (object server). Each of these items is moved individually over to Tivoli Storage Manager for long-term storage. As a result, there are 100 entries in the Tivoli Storage Manager database and 100 separate items in Tivoli Storage Manager storage. Each entry in the Tivoli Storage Manager database is about 600 bytes on average. Because Version 8.3.1 Content Manager offers a feature for aggregation of objects when using SSAM server.

Restriction: This feature is not available when using Tivoli Storage Manager instead of SSAM.

Many objects get aggregated to one big storage object which is then written to the SSAM server with this feature. It is not using the "object aggregation" of SSAM. IBM Content Manager has its own object aggregation algorithm.

At a high level, IBM Content Manager uses the Resource Manager migrator when constructing the object aggregate. The Resource Manager migrator determines if the source storage class is configured for object aggregation and if the target storage class has SSAM volumes. The migrator will then create object aggregates when moving data from source storage class to target storage class.

We do save significant space in the TSM Database with this new CM object aggregation feature. For every aggregate object only one entry will be written into the SSAM database. The complete aggregated object with all parts will be obtained during the retrieve operation.

Content Manager OnDemand, on the other hand, uses storage objects whose default size is 10 MB. CommonStore puts all e-mails that go into the same application group together in such a storage object. In this example, all e-mails (amounting to a total of 8 MB) are stored in just one storage object in the Content Manager OnDemand cache. For long-term storage, this one storage object is periodically migrated to Tivoli Storage Manager. Consequently, there is only one entry in the Tivoli Storage Manager database and just one item in Tivoli Storage Manager storage.

Tivoli Storage Manager's primary purpose is the backup and restore of individual files. CommonStore, however, requires storing some additional information beyond the file name. This is the reason that for each archived e-mail, CommonStore creates two items in Tivoli Storage Manager: One item holds the e-mail, the second, very small item the additional, CommonStore internal information.

Table 9-4 summarizes the previous analyses of the different storage concepts of Content Manager, Content Manager OnDemand and Tivoli Storage Manager.

Table 9-2 E-mail archiving example to illustrate the different storage concepts

Backend repository	CM without object aggregation	CM with object aggregation - only with SSAM	CMOD	Tivoli Storage Manager
# of e-mails in MS Exchange	100	100	100	100
# of items in archive	100	1	1	n/a
# of TSM database entries	100	1	1	200
Size of all TSM database entries	75KB	0.75KB	0.75KB	150KB
# of TSM storage items	100	1	1	200

Bear in mind that this example does not take into account any duplicate storage within Tivoli Storage Manager. This is usually done to increase data security. If one storage media is destroyed or gets corrupt, Tivoli Storage Manager can access automatically the copy of the archived e-mail in a different storage pool (on a different media).

The low number of items in Tivoli Storage Manager can be regarded as a particular strength of running CommonStore with Content Manager OnDemand, especially in large archiving projects where several million documents are archived every year. Fewer entries in the Tivoli Storage Manager database make the daily operation more efficient, because the Tivoli Storage Manager database becomes smaller and its backup (and also restore) becomes faster. Content Manager has a similar feature since Version 8.3.1 when using an SSAM server as storage management server.

Therefore, when considering Content Manager in large archiving projects where several million documents are archived, you either implement several Content Manager Resource Manager with different Tivoli Storage Management server connected or several Tivoli Storage Management server connected to one Content Manager Resource Manager to distribute the request and objects. Also an environment with Content Manager, SSAM server, and object aggregation is an option.

Metadata (attributes)

Both Content Manager and Content Manager OnDemand allow storing metadata (attributes) together with each archived object because both repositories are based on a relational database. But you cannot store any metadata in Tivoli Storage Manager due to the lack of such a database.

In the case of e-mail archiving, for instance, such metadata could be the subject or sender field of the message. For each attribute, a specific data format has to be selected that matches the data format in the business application. See Table 9-3 for an overview of how to map the different data formats in the business application to attributes in Content Manager or Content Manager OnDemand. Note specifically that a timestamp is stored as a variable string in Content Manager OnDemand. In the case of a text variable, the maximum length in Content Manager OnDemand is 254 characters compared to just fewer than 32,000 characters in Content Manager.

Table 9-3 Mapping attribute formats

business application	CM	CMOD
Text	Variable character, extended alphanumeric	Variable string, mixed case
Number	Integer, long integer or decimal	Integer, small integer or decimal
Date only	Date	Date
Time only	Time	Time
Date and time	Timestamp	Variable string

There is another interesting difference in how Content Manager OnDemand manages the attributes compared to Content Manager. In Content Manager OnDemand, old attributes can be migrated to Tivoli Storage Manager and stored on tape. This optional feature might be of particular value when dealing with very long data retention requirements or very large volumes. Content Manager and Content Manager OnDemand also differ in the maximum number of attributes, but even the lower one (32 with Content Manager OnDemand) is definitely high enough when used with CommonStore.

In addition to the application-related attributes, some additional technical attributes have to be set. As an examples: for an enhanced security during retrieval, CommonStore for Lotus Domino requires two additional technical attributes:

- ▶ CSLDOrigUser
- ▶ CSLDOrigDB

These security attributes have to be configured in both Content Manager OnDemand and Content Manager as separate attributes. During reload into the Domino database, CommonStore for Lotus Domino compares the actual values of the Notes environment (replica ID of the database where the document is restored to, Notes user requesting the retrieval) with the stored security attributes. If they do not match, the restore request is not fulfilled and the job goes into error.

Compression

Compression is of particular importance for e-mail archiving where very high data volumes are off-loaded. E-mails usually contain a lot of line data (or attachments with line data) that can be compressed considerably. The average compression rate that we have seen in messaging environments is about 50%.

Compression does not only save storage space, it has also a very positive impact on retrieval performance if tape or optical storage is used: The more data on one medium, the fewer media changes are required for retrieval. Because the automatic insertion of the medium into the drive consumes the most time at retrieval, the average retrieval time can be significantly lowered. This allows also keeping the number of parallel drives low.

Let us now have a look at the different backend repositories when using them with CommonStore for Lotus Domino or CommonStore for Exchange.

Content Manager has no built-in compression but can use the Tivoli Storage Manager client-based software compression. As a result, the storage space within Tivoli Storage Manager is smaller as within Content Manager.

Content Manager OnDemand comes with a very efficient built-in compression mechanism. The compression rate is very similar to ZIP and is on average 50% for e-mails. Because

compression is already done on the Content Manager OnDemand Server, there is no requirement to turn on Tivoli Storage Manager software compression. The storage size within Tivoli Storage Manager equals the one within Content Manager OnDemand.

CommonStore can use the Tivoli Storage Manager software compression as well. This option can be activated within the archint.ini configuration file.

Tivoli Storage Manager comes also with a so-called hardware compression feature that is supported in conjunction with selected tape drives. If such a device is available, the compression applies for Content Manager, Content Manager OnDemand and Tivoli Storage Manager backend repositories.

Retrieval and search options

In some CommonStore projects, you have the requirement to make the archived items searchable. Consequential all archived items must be stored together with metadata in the backend repository.

There are several clients which you can use to search for items archived with CommonStore. It is possible to search from the mail clients such as Notes or outlook, from SAP GUI with the help of SAP Document Finder as well as from backend archive clients such as Content Manager windows client (pClient), Content Manager OnDemand windows client, Web client (eClient), and Document Manager client.

The Content Manager repository comes with comprehensive search features for the metadata. In addition, full-text indexing and search has been tightly integrated into the base product since Version 8. There are two types of full-text options:

- ▶ A full-text index on the attributes (metadata)
- ▶ A full-text index on the content of archived items

Note: Content Manager OnDemand and Tivoli Storage Manager as the content repository do not support full-text indexing and searching.

The full-text index of content can only be generated if the data format of the archived item is supported by the filter (outside-in technology by Stellent). These search capabilities can also be used for items that have been archived by CommonStore. As an example, assume that several file attachments with different extensions such as DOC, PDF, GIF, JPG, and PPT were archived by CommonStore. The Content Manager full-text engine will update its index periodically and include the content of the DOC, PDF and PPT format. Graphic formats are automatically excluded.

When installing CommonStore with Content Manager special filter will be installed on the Content Manager server to support additional formats such as CSN, DXL, MSG for full-text search. This special filters ensures that the text portion of both mail body and file attachments are extracted and become part of the full-text index. This is important when using e-mail archiving.

Restriction: The full-text feature of Content Manager in combination with CommonStore is only available when using DB2 as the database for Content Manager.

Due to the usually huge volumes (several terabytes) in e-mail archiving, the full-text index can also become very large and in the range of several 100 GBs. Content Manager allows you to split up the indexes into several blocks that can be joined at search time. Nonetheless, the pure size of the full-text data causes additional cost in operation (maintenance, backup) and additional investment in hardware (storage, cpu).

The flexibility of Domino allows an alternative approach to make the full-text index more manageable. It is based on creating an abstract of each mail, limited to a maximum size. CommonStore for Lotus Domino is able to store this abstract as an attribute in Content Manager. If this attribute is enabled for full-text indexing, the user can do a full-text search on the abstract both from the Content Manager Client and in the Notes Client through the CommonStore for Lotus Domino search functionality. This approach might be an excellent compromise between optimizing the search results and keeping operational costs of the archive at a reasonable level.

Content Manager OnDemand has strong metadata search, retrieve and viewing capabilities but no pre-built full-text search capabilities. The search method is to find data in specific Content Manager OnDemand application groups. This makes it sometimes complicated to combine searches such as “search in e-mails *and* documents” (which have been scanned in or imported in a different way into Content Manager OnDemand).

Another option during archiving with CommonStore is to delete items in the source system when archiving them into the backend archive. Let us describe this by means of e-mail archiving. You can delete just attachments or you delete the e-mail including the attachment and leave only a header (called a stub) or you delete the whole mail from the mail system. When deleting the whole mail you will lose the link to the e-mail you archived. Therefore, if you are using a content repository such as Content Manager or Content Manager OnDemand you can search for this item based on the attributes and you can retrieve and restore it. This does not work with Tivoli Storage Manager as the archive.

Note: There is no possibility to search for archived items stored with Tivoli Storage Manager only as the backend repository. This is because Tivoli Storage Manager does not allow storing metadata together with the items.

Partial retrieve

With CommonStore for SAP, some very large objects are sometimes archived. This is particularly the case for some SAP print lists that can become larger than 1 GB. In order to optimize the retrieval performance, SAP builds indexes that are used to access a subset of a document directly. SAP then requests CommonStore for SAP to retrieve a certain data length at a certain data offset in a document. This is also called “partial retrieve”. As a result, the user can access the document content much faster.

Content Manager together with Tivoli Storage Manager allows partial retrieve. If the requested document has been moved from Content Manager to Tivoli Storage Manager, though, Content Manager will retrieve the entire document from Tivoli Storage Manager. Therefore, the partial retrieve of 50 KB by CommonStore for SAP might nonetheless result into moving 1 GB from tape (Tivoli Storage Manager) to hard disk (Content Manager).

Although Content Manager OnDemand has a built-in partial retrieve, it is of no value in the context of CommonStore for SAP because Content Manager OnDemand builds its own index for partial retrieve that is different from the one managed by SAP.

Tivoli Storage Manager is the only backend repository where CommonStore for SAP can do a partial retrieve. Because the software compression in Tivoli Storage Manager would make the offsets invalid, CommonStore for SAP suppresses Tivoli Storage Manager software compression during archiving.

SAP uses partial retrieval also for archived SAP data. SAP data archiving bundles and compresses data from many transactions into one item prior to offloading it by CommonStore for SAP. As part of this process, an internal index with the respective data offsets is created.

Although the objects themselves are not that large (10 MB), the partial retrieval provides a faster response time when the user wants to access a specific archived transaction.

Tip: Due to the technical differences outlined previously, we recommend that you use Tivoli Storage Manager as a back-end repository for CommonStore for SAP when archiving large print lists or doing SAP data archiving.

Encryption of mails

Notes comes with the option to encrypt mails. This means that only the sender and the recipients can access (decrypt) the mail body and the attachments. How does this affect archiving?

Because CommonStore cannot access the mail body (rich text) and the attachments (which are attached to the mail body) the only option is to archive the entire messages in CSN format (notes native). If stubbing is required CommonStore will remove the body and attachments and will leave an empty body with an link. CommonStore cannot create a “summary” of the mail body because it cannot access the mail body.

Also it is not possible to use the full-text search of mail content when archiving encrypted mails.

Single-instance-store (SIS)

This important archiving option is avoiding the storage of unnecessary duplicates of e-mails which in turn provides reduced storage consumption. A unique identifier (hash code) will be generated based on the message body and some other key message properties. CommonStore calculates the hash code for the e-mail and checks if this hash code already exists. It will store a link only (instead of archiving the e-mail) if it exists already. Therefore, if you send an e-mail to multiple recipients (with cc or bcc) and archive this e-mail, it is stored only once. This feature is only available when using Content Manager as the backend archive.

There are special considerations for CommonStore for Exchange. CommonStore for Exchange can archive mailboxes and PST files (Personal Stores).

If CM is used as backend repository and the CommonStore for Exchange single-instance-store (SIS) algorithm is activated to avoid duplicates of identical e-mails, CommonStore for Exchange will calculate a hash for each e-mail and check if the same e-mail is already stored in the archive.

This works great if the e-mails are archived from user mailboxes. The SIS algorithm, however, does not work if the e-mail is archived from a PST file. The reason is that Outlook/Exchange treats the message as modified because it is moved from a mailbox into a local PST file. Because the modified flag is set, CommonStore for Exchange calculates a different hash code. This results in a separate copy in the archive system, even if the e-mail appears to be identical.

Tip: Such features like full-text search or single instance store are quite common customer requirements with e-mail archiving. These requirements can be met when using Content Manager as content repository only.

Integration into SSAM

One of the options for archiving data with CommonStore is to store the data directly into Tivoli Storage Manager. CommonStore supports Tivoli Storage Manager based on the Tivoli Storage Manager archive API. Because of this it supports SSAM as well.

In order to work with an SSAM server, the Tivoli Storage Manager client sample file simply must have this parameter set: ENABLEARCHIVERETENTIONPROTECTION ON.

For more details on how to set up CommonStore with Tivoli Storage Manager, check the following publications:

- ▶ *IBM DB2 CommonStore for Lotus Domino Administrator's and Programmer's Guide Version 8.3*, SH12-6742-03
- ▶ *IBM Content Manager CommonStore for Exchange Server Administration and User's Guide Version 8.3*, SH12-6741-03
- ▶ *IBM DB2 CommonStore for SAP Server Installation and User's Guide Version 8.3*, SH12-6744-03

The main difference from Content Manager and Content Manager OnDemand is that CommonStore does not support sending events to SSAM. CommonStore uses the chronological retention to store data. Therefore, the retention period is declared when an object is stored. As stated previously, it is important to have retention definition of data synchronized in the application (SAP system or e-mail application) and within retention managed storage (SSAM).

It is possible (and recommended) to define retention periods in SAP systems in the same matter as defined in retention managed storage (SSAM).

But how about e-mail systems? Do they have retention management for e-mails on the application level?

There is no equivalent retention management in e-mail systems. This can implicate a synchronized environment in two ways:

- ▶ The e-mail itself (with links inside) gets deleted (by user) in the e-mail application although the referenced and archived data such as attachments did not expire in SSAM. In consequence you will lose the links to the data stored. There is a way to keep this in-sync. CommonStore can be triggered from the e-mail client to delete linked data when deleting the e-mail itself. This is not the default behavior and would be part of the customizing.
- ▶ The referenced and archived data (such as attachments) gets deleted in SSAM because the defined retention period expired. The links inside of these e-mails will not work anymore because SSAM will not find the linked data.

We recommend that you store e-mail data into Content Manager first and not directly onto SSAM to avoid such problems, and consider implementing a records management solution (to be discussed later).

Also consider the following features when using CommonStore for SAP with SSAM server. Within the SAPGUI, a user can add notes (remarks) to each archived document. This is a feature that is most frequently used in a SAP work flow where comments are added during the various processing steps. SAP notes can also be added to reports or other inbound or outbound documents. SAP notes are not available for SAP data archiving.

Storing these electronic notices on any device controlled by SSAM, including the DR550, is an issue. These devices have an additional layer of protection to provide WORM functionality and do not allow to update or delete these electronic notices.

If CommonStore for SAP is directly connected to SSAM or DR550, creating a SAP electronic notice will result in an error and is not supported.

If CommonStore for SAP is connected to Content Manager (and Content Manager is linked to a DR550), it is necessary to configure Content Manager in such a way that the electronic notes are stored in a different part item-types (ICMNOTELOGCS). The configuration of this additional part item-type is described in the CommonStore for SAP server manual. This approach ensures that the actual document (object) is archived on a DR550 in a compliant way, but any notes can easily be added because ICMNOTELOGCS is linked to hard disk storage (within Content Manager or standard Tivoli Storage Manager).

If CommonStore for SAP is connected to standard Tivoli Storage Manager (without WORM protection), creating SAP electronic notice will work fine because CommonStore for SAP can remove an old notice and add the updated one without problems.

Also consider the following points when using Content Manager OnDemand with Tivoli Storage Manager or SSAM attached as the backend archive for CommonStore:

- ▶ Only the Content Manager OnDemand expiration types “document” and “segment” of application groups are supported by CommonStore.
- ▶ Stored objects will not expire when using event-based retention of Tivoli Storage Manager or SSAM.
- ▶ Objects will get deleted in Content Manager OnDemand and Tivoli Storage Manager independently of each other (based on both retention definitions) when using chronological retention of Tivoli Storage Manager. OnDemand can not trigger Tivoli Storage Manager directly to delete objects when deletion or expiration of objects in OnDemand occurs (disconnected process).

In summary, there are several customer requirements for a specific solution such as e-mail or SAP archiving. These requirements must get priorities. Based on the prioritized requirements and the described recommendations, you can decide which back end solution fits best.

9.5 Records and retention management

Organizations have an obligation to comply with all laws, regulations, and standards which are applicable to their business activities. This includes satisfying requirements related to the creation, retention and disposition of records that are created or received in the regular course of business.¹

Records, with their respected accuracy, detail and completeness, have historically been regarded as the “corporate memory” – documenting daily business actions and decisions. When records meet both operational and legal requirements, they are recognized as the most trustworthy evidence of an organization’s voluminous transactions and processes. As such, records enable companies and government agencies to review, analyze or document the specifics of their past actions and decisions.

In recent years, due to the increasing quantity of litigation and necessity for regulatory compliance, records have assumed even greater value. The tide of regulatory changes and high profile litigation has raised the threshold and level of scrutiny for compliance. A growing body of laws, regulations, legal precedence as well as national and international best practice standards and guidelines have collectively established a set of common functional requirements for the protection and preservation of electronic records.²

¹ Cohasset Associates, White Paper for IBM, October 2004

² Cohasset Associates, White Paper for IBM, October 2004

Note: “A record is information created, received, and maintained as evidence and information by an organization or person, in pursuance of legal obligations or in the transaction of business.” (ISO 15489)

Records arise from business activities such as events, decisions, or transactions and tell the complete story about the activity. All records convey information.

A record must be what it purports to be. It must be a full and accurate representation of the transactions, activities, or facts to which it attests. A record must be complete and unaltered. Also a record must be able to be located, retrieved, presented, and interpreted.

Therefore, the main reason why companies introduce records management is to comply with all laws, regulations, and standards which are applicable to their business activities, to reduce risk of litigation and sanctions and to reduce legal costs. There are other reasons as well such as improving operational efficiency, better handling of changes in regulations and retention periods and of physical paper records.

A records management system is the guardian of records access, records circulation, and audit trails. Context information is used to determine individual's actions, authorization for the action, and the action date. The record of action provides evidence of abuse, misuse, and non-compliance with administrative and legal regulations.

This is different than content management. Content management provides the ability to capture, store, and manage content. Records management works within this type of infrastructure to apply formal, rules based, management to the retention and disposition of that stored content. These rules are based on regulations, laws, and business policies and might be unique for each and every organization. Records management is all about control — basically, making sure that you only keep what you have to keep for as long as you have to keep it and then afterwards making sure that it is destroyed.

Therefore, the difference between records management and retention management is in the decision-making process and the record keeping oversight and control, as shown in Table 9-4.

Table 9-4 Records and Retention Management differences

Records management	Retention management
Control of corporate “Records” (documents that require full record keeping control)	Control of “non-records”
Uniform Program and Processes for Paper and Electronic Records	Manages Electronic Records only
Formal Declaration of a Business Record, including central file plan based on Retention Schedule and Legal Holds and formal approved retention periods	No records declaration, the assignment of retention periods and destruction dates is based on predefined policies
Retention based on Time, Events or a combination of both	Time-Based Retention or Event-Based Retention (not the combination)
Flexible to change retention rules to reflect changes in Regulations and Retentions	Can lengthen retention period, but cannot shorten it after it is set.
Records-based Access Control and Security based on corporate-wide security model	Access Control and Security based on Content Management security model

Records management	Retention management
Provides extensive audit trail capability and reporting (who, what, when, why)	Limited audit trail capability, it tells when a document was destroyed
Deletion of records only by authorized records practitioner	Automatic deletion after expiration of retention periods

In Retention Management, there is no formality or centralized control and oversight. Anyone can assign arbitrary retention periods and destroy information at any time. There is a potential risk that we might not be applying the right retention period, or that we might be keeping something we should not, or destroying something we should be keep. Retention Management will only be as appropriate and as effective as the quality of decisions we apply to it, and how carefully we administer it. Retention Management does give organizations “some” degree of control to business documents where there might otherwise be none. Many information management software solutions offer this basic Retention Management capability.

As detailed in the Figure 9-14, retention and disposition management is only one part in the set of records management functionality typically required to meet mandatory and voluntary compliance requirements.

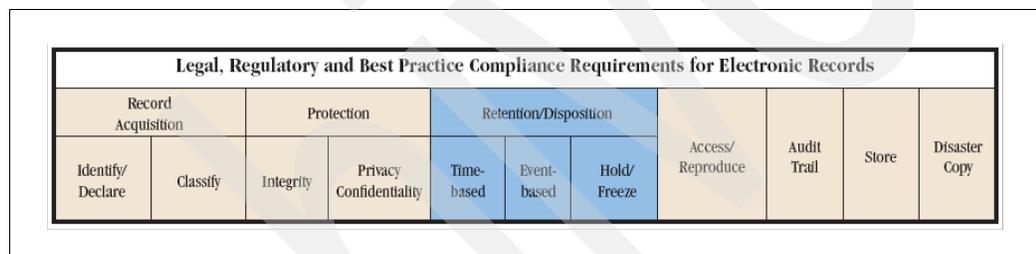


Figure 9-14 The general set of records management functions

Records Management, utilizes a centralized process for acquisition, protection, retention and disposition of records. Only an authorized records practitioner can destroy records and only when retention periods have expired and there are no legal holds. At the end of the day, Records Management, properly administered, gives organizations full accountability.

The purpose of placing content under records management control is to ensure that:

- ▶ The integrity of the content is protected.
- ▶ The record is protected against deletion and alteration.
- ▶ The record is accessible during the retention period.
- ▶ The lifecycle rules for the record are properly managed.

In order to place content under records management control, the content must be declared and classified according to the lifecycle management rules of the Corporate File Plan (CFP). The compliance requirements for integrity protection, retention and accessibility of records can only be achieved through correct and consistent declaration and classification.

Corporate File Plans and Retention Rules help records managers translate regulations into specific records retention periods. Using a retention rule applied to a specific type of record or information asset helps organizations ensure proper records retention periods. For instance, assigning a 3-year fixed retention rule to information ensures that it will not be deleted before the 3-year period is up. Proper review of upcoming expirations also helps organizations make sure that the records are properly disposed to eliminate or minimize the liability from retaining it too long.

Records management enforcement

Records management mainly enforces the following three actions:

1. Declaring a document to be a corporate record containing content critical to the business: Puts a document under records management control (either automatically using related metadata or manually by user). The user can no longer edit or delete the item. The records manager has the exclusive ability to edit and delete. Records management related metadata is applied to the item
2. Classifying that record applies the appropriate retention rule to the record based on subject and/or content type. This rule can be assigned manually or via auto-classification policies
3. Applying lifecycle management: this ensures that the record is available until the end of the retention period and it destroys or transfers the record out (for example, to federal archives) according to assigned rules.

Any other action is a non-record keeping consideration. Corporations must have a Records Management solution that is not additive to the normal work flow process, that does not require replication or migration of the document content being treated as a record and that will address every type of electronic or paper document.

With the staggering volume of electronic records being generated in today's organizations, the single greatest challenge (and the most likely shortcoming) of an electronic records management environment is ensuring that the declaration and classification of records do not create an unacceptable burden on the individuals that use the applications and systems.

One key to successful records management is deciding, up front, what information is to be kept, and for how long. If you retain too little you might be facing potential regulatory issues, fines. If you retain too much or for too long you might expose yourself to a potential liability long after that information could have been legally and properly disposed.

Retention periods

There are two types of retention periods:

- ▶ Time-based also called "fixed term retention," the time-based retention period begins at the time the record is received, created or possibly stored, and ends after a predetermined period of time (for example, the date of the business transaction that spawned the record plus six years, or the date the record was declared plus three years).
- ▶ Event-based although the retention period begins at the time the record is received, created or possibly stored, the total period of retention is variable and consists of two parts:
 - An indeterminate amount of time until a predefined "event" occurs (for example, until the closing of an account, payout of an insurance policy, payoff of a loan, termination of an employee, and so on.) plus
 - A predetermined period of time after the event. (for example, three years after the closing of the account).

Time-based retention is relatively easy to manage because the start date is when the record is declared and classified (for example, received, created, or stored) and the total retention period for the record is known in advance.

Event-based retention is comprised of an undefined “pre-event” retention time and a predefined “post-event” retention time that, when added together, constitute the total event based retention period. It is typically more complicated to manage because the system, application or person managing the record must be made cognizant of a “trigger” event that starts the post event retention period. The responsibility for communicating the “event” generally resides with the business application or business function that has responsibility for the process and records in question.

Retention periods govern the length of time a record must be kept unless there is a legal or regulatory action (or audit) which requires that certain records be “held,” “frozen” or suspended” from possible destruction for an indefinite period of time – until the conclusion of the special “triggering” activity/event. This is called “legal hold” of electronic records.

Management for paper records

Another important area where records management differs from retention management is the management of paper documents. It is a common requirement that the underlying record keeping infrastructure and processes must be applied to both electronic records and paper records.

The paper records are identical to the electronic records with the exception that there is no content stored electronically in the system only metadata. These records are managed and tracked within the system with unique identities and barcoding technologies for check-in and check-out.

Important: Developing a compliant records management environment requires more than just products or technologies. Compliance is a process.

9.5.1 DB2 Records Manager integration into DB2 Content Manager

The architecture of Records Manager was designed to support the separation of records management administration and disposition control from the location of record content. Records Manager provides an engine for controlling the disposition and management of the lifecycle of electronic records (as well as physical records) as described in 3.5. The Content Manager is the content repository where the record content and related search metadata is stored.

The Records Manager and Content Manager architecture, depicted in Figure 9-15, provides an integrated platform for declaration and classification, storage, retrieval, disposition control and lifecycle management capabilities for all records regardless of whether the records were created or acquired by either e-mail or by office applications such Microsoft Office.

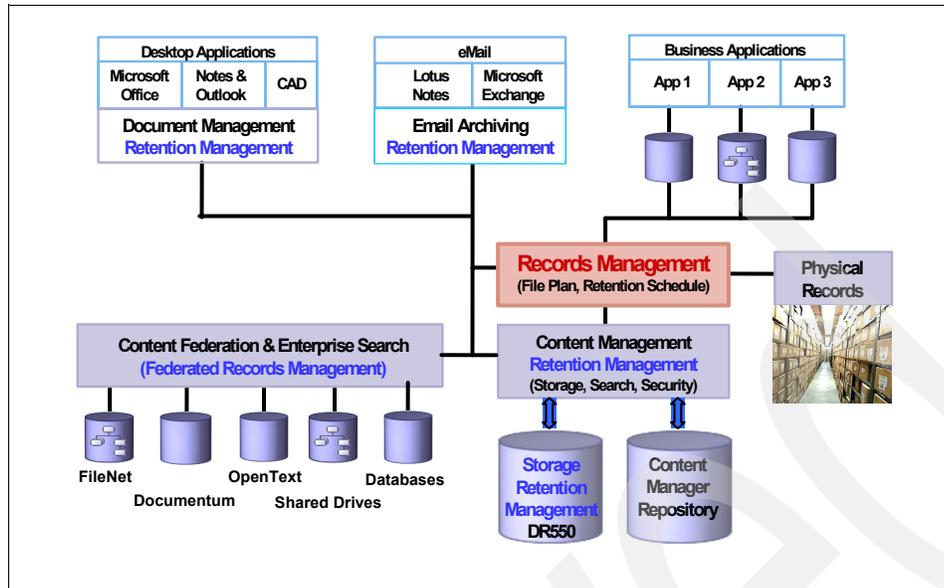


Figure 9-15 IBM Records Management Solutions Architecture

Record-enabled products are:

- ▶ IBM DB2 Content Manager
- ▶ IBM DB2 Document Manager
- ▶ IBM DB2 CommonStore for Lotus Domino
- ▶ IBM DB2 CommonStore for Exchange Server

Note: CommonStore for SAP is not enabled for DB2 Records Manager.

Declaring and filing records with Content Manager as the repository

With the introduction of Records Manager and its integration with Content Manager, Records Manager became an alternative for controlling the disposition of data within Content Manager.

Records Manager provides access control for all functions related to records administration. At the time of declaration, DB2 Content Manager establishes DB2 Records Manager as the only authorized “user” to access, change or deletion data. This is accomplished by executing the following steps:

1. Content Manager removes all prior user authorizations for document manipulation or deletion.
2. Content Manager sets up Records Manager as the singular “user” authorized to initiate the deletion of the record.
3. Content Manager notifies Records Manager that deletion control has been properly transferred.
4. Security authorization between Records Manager and Content Manager are synchronized so that any security authorization changes that occur in Records Manager will automatically be reflected in Content Manager.

Within the record-enabled solution, Records Manager is configured to override Content Manager native access privileges for declared records – a key requirement of both DoD 5015.2 Chapter 4 and UK National Archives. At search time, security permissions for declared records defined by Records Manager prevail over permissions within Content Manager.

After these steps have been accomplished, no other user with access rights to Content Manager can alter or delete the declared record. An audit trail of activities against the declared record is maintained by Content Manager and by Records Manager. Records Manager can be configured to extend its auditing with the audit data recorded by Content Manager throughout the record's life. This "combined" auditing goes beyond the DoD 5015.2 and UK National Archives 2002 audit requirements.

Following classification of the record and assignment of the appropriate retention period to the record, Records Manager controls the final disposition of the record based on the assigned retention period, or any extension of the period do to a legal or regulatory hold.

Deletion of declared records with Content Manager as the repository

After Content Manager has declared a record to Records Manager and it has been classified, all disposition actions related to the record, including time or event-based and the setting and releasing of legal or regulatory holds, are handled by Records Manager.

The final disposition of a record is based on the lifecycle rules contained in Records Manager as part of the definition of the records category or series in the file plan. Records whose retention period has expired are identified by records manager (a person - not the application) and managed through a process of approval and final deletion that includes checks and balances such as determining whether a hold is in place for the record in question.

Because Records Manager is the only authorized "user" capable of initiating the deletion of a record where the record content is stored and managed on Content Manager, the deletion of a record is an interactive process between Records Manager and Content Manager.

The deletion of a record involves the following general steps:

1. Records manager identifies records where the retention period has expired.
2. Records manager checks to ensure that no hold order or suspension of the records is still in effect.
3. Records manager creates a report (electronic or paper) that can be used for review or to notify the appropriate parties, such as the business owner, legal and compliance, to conduct a review of the records pending deletion and indicate their approval or denial (with reasons).
4. After the records have been approved for deletion, Records Manager creates a list of records to be deleted and sends the list to Content Manager.
5. Content Manager deletes the content of the records, including all links and all related metadata.
6. Content Manager confirms to Records Manager that the deletion has been successfully completed.
7. Records Manager deletes the metadata for the records and the disposition process is complete.
8. Both Records Manager and Content Manager retain complete, detailed audit trails of the actions that have been taken to delete the records, thereby completing the "chain of custody" for the records.

Through this comprehensive process with documented audit trails that are retained as records, it can be demonstrated that the deletion of records was accomplished in a reliable and trustworthy manner.

9.5.2 DB2 CM and Storage Management together with DB2 Records Manager

Most regulations simply authorize the storage of specified records on an alternative media – optical storage, magnetic storage, or microfilm. Some regulations, however, are explicit regarding not just the type of media, but also the functionality of media and how the records are recorded on the media. Although relatively few in number, the agencies promulgating these specific storage regulations are among those with the greatest regulatory visibility. They include: the Securities and Exchange Commission (SEC), the Commodity Futures Trading Commission (CFTC), and the Nuclear Regulatory Commission (NRC). The electronic records management regulations from these agencies explicitly call for the use of a non-rewritable, non-erasable information recording process.

Because all regulations require the protection of a record's integrity, there is an implicit requirement to provide a substantively equivalent “non-rewritable, non-erasable, non-alterable” management environment, whether by the content and/or records management application, by the storage management software, by storage hardware, or a combination of these.

Choosing the right components to provide the equivalent “non-rewritable, non-erasable, non-alterable” management environment depends on the business requirements.

Basically, in an environment where Records Manager is integrated in Content Manager, Records Manager controls the access, changes and disposition of the documents.

Note: There is no interaction between Records Manager and Tivoli Storage Manager / SSAM. Records Manager interacts with Content Manager and Content Manager with Tivoli Storage Manager / SSAM.

Integrated filing of records

Information objects can be declared as company records with the help of record-enabled products as described previously. Some of these information objects will be declared on creation time. Other documents have a lifecycle with different states before declaration time such as “creation”, “modification and versioning”, and “approval” for example with the Document Manager Integration.

- ▶ If a Tivoli Storage Manager server is connected all documents will be stored into Content Manager without any expiration set in Tivoli Storage Manager. The Content Manager retention value for item types can be set to forever, because it will be overwritten by the retention rules of Records Manager.
- ▶ If an SSAM server is connected all documents will be stored with an undefined retention time into the SSAM server. We recommend that you set RETMIN to 0 if compliance is not required or to the minimum supposed retention period in environments where compliance is required. Set RETVER=0 because the Content Manager metadata was deleted already at this point of time. The parameters RETMIN, RETINIT and RETVER will be set in the configuration of the appropriate SSAM or Tivoli Storage Manager management class. The Content Manager retention value for item types can be set to forever, because it will be overwritten by the retention rules of Records Manager.

Again choosing the right components to provide the equivalent “non-rewritable, non-erasable, non-alterable” management environment depends on the business requirements. The integration of Content Manager and Records Manager with Tivoli Storage Manager or SSAM server can be sufficient in some cases. An integrated solution with IBM DR550, on the other hand, is certified by KPMG and offers additional security and protection such as restricted access for root users, restricted access for Tivoli Storage Manager administrators and restricted Tivoli Storage Manager functionality.

Also keep in mind that a solution with SSAM server only (without DR550) requires an additional SSAM server license.

Integrated deletion of records

Normally the disposal of records is a process initiated within DB2 Records Manager. A records manager (a person — not the application) runs a process whereby DB2 Records Manager will present the records manager with the list of records due for disposal. This calculation is based solely on the information contained in DB2 Records Manager. After the records manager has reviewed the list, they click **Proceed**. This starts a batch process whereby DB2 Records Manager initiates a series of calls to Content Manager to delete the records. Content Manager will then delete the index rows for the documents and issue a delete command, using the Tivoli Storage Manager API to delete the documents.

- ▶ If a Tivoli Storage Manager server is connected simply a Tivoli Storage Manager API delete call will be issued.
- ▶ If an SSAM server is connected a Tivoli Storage Manager API “event” will be issued.

DB2 Records Manager awaits for a response from Content Manager with the “success” or “failure” for each delete attempt. If it receives a success response, it removes the pointer metadata to the record in Content Manager and writes the deletion in the lifecycle processing log.

If a record is under a legal hold in DB2 Records Manager, the record will not even appear as eligible in the first place. Also, any attempt to perform an ad-hoc deletion of the record in DB2 Records Manager would fail.

Note that after something is declared as a record, there is no way to delete it from within Content Manager (or other application). You can only delete it through DB2 Records Manager which then in turn issues a Delete API call to Content Manager with a special privilege to complete the deletion.

9.5.3 Use cases for the described configurations

This section describes some example configurations and scenarios for a better understanding of the interaction of the products and the resulting behaviors. It assumes that the whole stack of products is fully integrated.

Sample scenarios

Here are some typical scenarios:

1. Documents stored for x amount of time with retention period started at creation date.

This scenario shows objects stored for an amount of time such as 5 years. In a non-records managed solution, the objects will not be deleted throughout the stack automatically. In a records managed solution, the records manager is the only component to initiate the delete and logs all such requests and completions.

2. Documents stored for x amount of time with retention period starting with specified event for example, account close.

This scenario assumes that the documents would be stored initially forever waiting on an event such as the closure of the account to determine the retention period after the closure.

In a non-records managed system, this event would have to be monitored either with a custom application or by a person. In either event significant work must be undertaken. A metadata field could be defined for the event. Someone or something must fill out this field and monitor it. After some period time calculated from the value in the field or filling out of

this field someone or something would issue a delete. This person or program must be fully aware of retention rules for the object and must also record the rules, the time period and the deletion request and completion. In a records managed system this is business as usual and part of the functionality of the product.

3. Retention period for stored documents is extended indefinitely due to other reasons such as document under legal investigation. This is called a legal hold or suspend.

In a non-records managed solution there is no allowance for this functionality. Just like the prior scenario, we could assume a “metadata” field were used to define this. You would be able to specify a date and/or a time to legal hold. After the legal hold was over you would have to determine whether you wish to delete immediately or continue down the original held time frame or extend it? In any event someone/some program will have to figure out the rules and how to apply them and how to record the actual event that happened to support the hold and then when to delete it later. This could be an administrative nightmare and would require significant programming effort to accomplish and test. In a records managed solution this is part of the functionality.

Rules for scenarios

The deletion could be handled through a custom application, or via Records Manager. Records Manager or any other application could initiate the delete assuming an authorized UserID has been used, and the document is not held. No unauthorized deletion by anyone else. The scenarios will be discussed only with Content Manager (not with Content Manager OnDemand) because the Records Manager integration is available with Content Manager only.

For the configurations without Records Manager, let us assume that there is a small application that checks the expiration date stored for objects in Content Manager. Authorized users could use such an application to search for the expired documents, generate reports (to get management approval, for example) and subsequently delete the expired items.

The SSAM server is configured with the following retention definitions: RETINIT = EVENT, RETMIN = x years and RETVER = 0 (Example 9-2).

Example 9-2 RETVER setup

```
define copygroup testdom testset testclass_chrono standard type=archive retver=0  
retinit=EVENT destination=archivepool
```

When using the normal Tivoli Storage Manager server, Content Manager only supports the Tivoli Storage Manager backup API and can use consequently Tivoli Storage Manager backup copy groups only. Backup copy groups do not have automated expiration processing for primary objects. The data in a backup copy group will only expire when an application (such as Content Manager) issues a Tivoli Storage Manager API call to delete the data. Therefore, there is no retention definition for primary objects stored within backup copy groups.

Table 9-5 Configurations and scenarios

Configuration	Time-based retention (documents stored for x years starting at ingestion date)	Event-based retention (documents stored for x years waiting for an event)	Retention period for stored documents is extended by a legal hold
<p>CM with x years retention</p> <p>+ Tivoli Storage Manager standard</p>	<p>Documents stored in CM will get an expiration date based on the retention definition of the CM item type (ingestion date + x years). A simple application could be used to search for expired documents and to delete them after approval. Objects stored in Tivoli Storage Manager will expire when the application (and CM as well) issues the delete call.</p>	<p>The expiration date for objects in CM is calculated based on the ingestion date (ingestion date + x years). Therefore, events are not observed. Therefore, this cannot be done without Records Manager or an equivalent application.</p>	<p>The expiration date for objects in CM is calculated based on the ingestion date (ingestion date + x years). Therefore, legal holds are not observed. Therefore, this cannot be done without Records Manager or an equivalent application.</p>
<p>CM with forever retention</p> <p>+ Tivoli Storage Manager standard</p>	<p>Documents stored in CM with forever retention policy do not expire. Consequently no object stored in Tivoli Storage Manager expires. Therefore, this scenario cannot be done without Records Manager or an equivalent application.</p>	<p>Documents stored in CM with forever retention policy do not expire. Consequently no object stored in Tivoli Storage Manager expires. Therefore, this scenario cannot be done without Records Manager or an equivalent application.</p>	<p>Documents stored in CM with forever retention policy do not expire. Consequently no object stored in Tivoli Storage Manager expires. Therefore, this scenario cannot be done without Records Manager or an equivalent application.</p>
<p>CM with x years retention</p> <p>+ SSAM server (add. license required)</p>	<p>Documents stored in CM will get an expiration date based on the retention definition of the CM item type (ingestion date + x years). A simple application could be used to search for expired documents and to delete them after approval. Objects stored in SSAM will expire when the application issues the delete call and CM the event call.</p>	<p>The expiration date for objects in CM is calculated based on the ingestion date (ingestion date + x years). Therefore, events are not observed. Therefore, this cannot be done without Records Manager or an equivalent application.</p>	<p>The expiration date for objects in CM is calculated based on the ingestion date (ingestion date + x years). Therefore, legal holds are not observed. Therefore, this cannot be done without Records Manager or an equivalent application.</p>
<p>CM with forever retention</p> <p>+ SSAM server (add. license required)</p>	<p>Documents stored in CM with forever retention policy do not expire. Consequently no object stored in SSAM expires. Therefore, this scenario cannot be done without Records Manager or an equivalent application.</p>	<p>Documents stored in CM with forever retention policy do not expire. Consequently no object stored in SSAM expires. Therefore, this scenario cannot be done without Records Manager or an equivalent application.</p>	<p>Documents stored in CM with forever retention policy do not expire. Consequently no object stored in SSAM expires. Therefore, this scenario cannot be done without Records Manager or an equivalent application.</p>

Configuration	Time-based retention (documents stored for x years starting at ingestion date)	Event-based retention (documents stored for x years waiting for an event)	Retention period for stored documents is extended by a legal hold
CM with x years retention + Records Manager + Tivoli Storage Manager standard	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including retention time, access rights and issuing the deletion of objects. Objects stored in Tivoli Storage Manager will expire when Records Manager (and CM as well) issues the delete call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including the lifecycle of objects (for example, "wait for event") Objects stored in Tivoli Storage Manager will expire when Records Manager (and CM as well) issues the delete call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control. Also authorized users can put a legal "hold" or "release" on objects. Objects stored in Tivoli Storage Manager will expire when Records Manager (and CM as well) issues the delete call.
CM with forever retention + Records Manager + Tivoli Storage Manager standard	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including retention time, access rights and issuing the deletion of objects. Objects stored in Tivoli Storage Manager will expire when Records Manager (and CM as well) issues the delete call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including the lifecycle of objects (for example, "wait for event") Objects stored in Tivoli Storage Manager will expire when Records Manager (and CM as well) issues the delete call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control. Also authorized users can put a legal "hold" or "release" on objects. Objects stored in Tivoli Storage Manager will expire when Records Manager (and CM as well) issues the delete call.
CM with x years retention + Records Manager + SSAM server (add. license required)	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including retention time, access rights and issuing the deletion of objects. Objects stored in SSAM will expire when Records Manager issues the delete call and CM the event call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including the lifecycle of objects (for example, "wait for event") Objects stored in SSAM will expire when Records Manager issues the delete call and CM the event call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control. Also authorized users can put a legal "hold" or "release" on objects. Objects stored in SSAM will expire when Records Manager issues the delete call and CM the event call.
CM with forever retention + Records Manager + SSAM server (add. license required)	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including retention time, access rights and issuing the deletion of objects. Objects stored in SSAM will expire when Records Manager issues the delete call and CM the event call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control including the lifecycle of objects (for example, "wait for event") Objects stored in SSAM will expire when Records Manager issues the delete call and CM the event call.	Retention stored in CM is not used in this scenario. Instead Records Manager has the control. Also authorized users can put a legal "hold" or "release" on objects. Objects stored in SSAM will expire when Records Manager issues the delete call and CM the event call.

The DR550 WORM solution could be used in all configurations described with SSAM server in Table 9-5. There are no changes to the scenarios when using DR550 because the SSAM server is part of this solution as well. The main advantages when using DR550 are:

- ▶ The DR550 solution is certified by KPMG.
- ▶ It is a pre-installed, pre-configured compliance box for WORM storage, including maintenance.
- ▶ There are pre-designed and tested high viability features available for DR550.
- ▶ It has additional security implemented on the base operating system level.
- ▶ The hardware frame, including the door, protects against unauthorized access.
- ▶ All software licenses for this solution are included.

Finally, let us consider possible changes to the retention configuration in the SSAM server and their implication. What would happen in the described SSAM scenarios if:

RETMIN > x years: the “event” call to delete data will be issued before RETMIN expires. SSAM will store the receipt of the “event” call in this case and will delete the data after RETMIN expires depending on the value RETVER.

RETMIN < x years: no change

RETVER > 0: If RETVER is bigger than 0, data will be kept x days longer (RETVER=x) in the SSAM server until it is deleted by the SSAM server. The metadata for this data (in Content Manager and Records Manager) is already deleted at the time the “event” call was sent. This could be useful for authorized users or administrators to have the possibility to access the data over a defined transition period.

File system archiving and retention

This chapter describes file systems and their relationship with ILM practices and retention management of data. We discuss solutions to move files between storage tiers and to retain them when required. We discuss:

- ▶ File systems and ILM
- ▶ Combining archiving with DR550 / SSAM
- ▶ Overview of TRIADE TriFSG software
- ▶ Solutions that benefit from this combination
- ▶ SSAM archive client
- ▶ Hierarchical storage management
- ▶ Tivoli CDP
- ▶ GPFS
- ▶ N series SnapVault or LockVault

10.1 File systems

Filesystems are a common and widely used and understood metaphor for ordering and accessing information. Many applications are capable of storing data as files in a file system structure that generally resides on a disk storage device. What kind of ILM services are applicable to a filesystem? We will consider two possible services to address business problems:

- ▶ Reduction of disk space required to store files
- ▶ Use of a file system to store data in a non erasable format for compliance

Reduction of disk space required can be achieved with Hierarchical Storage Management (HSM) techniques and products. HSM allows for data to be moved to a different storage tier based on policies such as age, size and total filesystem space. The application *sees* the data on the original filesystem, even though it has been moved to another storage tier. When the application references the data a transparent recall operation is automatically performed by the HSM system.

The second business problem is to store data on a filesystem in a non erasable way, probably for compliance reasons. There are various ways of doing this and we will illustrate two examples:

One is a software only solution based on Tivoli Continuous Data Protection (TCDP). TCDP allows for the creation of special protected folders on a workstation or server running Windows. TCDP protects data in specific folders according to predefined chronological retention policies. It does not permit the files to be deleted until the retention policy has expired.

The second approach allows the use of a DR550 solution as a file system with the TRIADE TriFSG DataGateway product from TRIADE. Using this gateway the application sees and accesses a network filesystem, using protocols such as CIFS and NFS. The TRIADE TriFSG DataGateway then stores the data onto the DR550 solution.

10.2 Archiving and retention

Data archiving is often performed for one of two reasons, or both at the same time.

The first reason is that the data is no longer required on online high performance storage devices because it is no longer expected to be used. The archival function can be used to move the data to lower cost storage devices, and move means the data is removed from the primary storage device.

The second reason data is archived is to create a copy of the data on non erasable, non rewriteable storage devices, often termed WORM (Write Once Read Many) devices.

ITivoli Storage Manager offers an archive function and this function can be used both for archiving files and for direct program integration. We now discuss the use of the archive function for archiving files.

10.2.1 The archive client

ITivoli Storage Manager offers file system data archiving in the standard Tivoli Storage Manager backup and archive client. Backup and archive are two completely different Tivoli Storage Manager concepts.

The backup concept

The backup concept refers to creating a second, backup, copy of data on a primary storage device. The backup is used if the primary data is unavailable because of storage hardware failures or application errors. Backup implies concepts such as versioning, backup copies taken at regular intervals, copies that are then expired quite soon. Backup is an operational process because it is used to guarantee application or recovery in the case of data unavailability. The backup process can be automated, the Tivoli Storage Manager backup client automatically determines which files to backup when it performs incremental backups, only new and changed files are backed up.

Some installations think that backup can be used as an archive, often we hear requirements for backup rotation schemes such as keep daily backups for a week, weekly backups for a month, monthly backups for a year, and so on. The reasoning behind the scheme is to be able to maintain very old copies of data without consuming too much storage space.

Now consider the following example: you have the backup tape for December 31 2002 and the tape for December 31 2003. You are protected and have historical data. You are now required to produce data that was on the system in May 2003. You restore both backups, one after the other, but you do not find the required data. The data was created in March 2003 and then deleted in August 2003. Your historical records scheme has not been able to give you the required data, therefore it is clearly flawed.

The archive concept

Tivoli Storage Manager offers a separate function, part of the standard Tivoli Storage Manager backup and archive client, to manage data that must be retained for whatever reason. This is the Tivoli Storage Manager archive client. The archive client allows you to archive files to Tivoli Storage Manager. Each archived file is assigned a retention, based on the management class and an initial destination storage pool. Retention can be either chronological or event based as discussed in Chapter 4.1, "Tivoli Storage Manager concepts" on page 74.

Data archival is not automatic, as is often the case in backup operations, but is initiated by the user. The user is responsible for deciding what to archive and for how long to keep the data after it has been archived. The process can be automated, but it is still the user's responsibility to decide what to archive and for how long.

To perform an archive the user or application calls the Tivoli Storage Manager archive client and specifies some or all of the following information:

- ▶ The name or list of names of files that have to be archived.
- ▶ The management class with the desired retention attributes.
- ▶ Whether to keep or delete the file from primary storage
- ▶ A description for the archive that you can then search on

The following example in Figure 10-1 contrasts a manual archive process with the Tivoli Storage Manager archive client functionality.

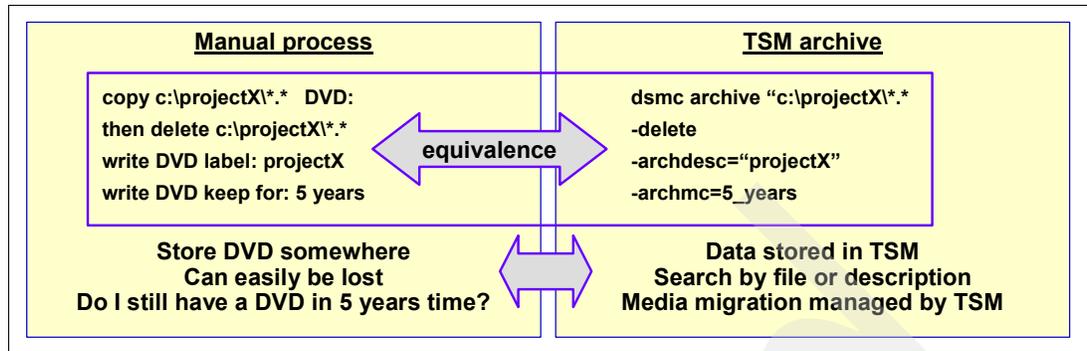


Figure 10-1 Contrasting Tivoli Storage Manager archive and manual archive

The Tivoli Storage Manager archive client is very simple to use and can easily be integrated into other software packages and programs so that they can exploit Tivoli Storage Manager storage and storage management functions.

10.2.2 Archiving and the SSAM and DR550

Archiving files to a retention protection enabled Tivoli Storage Manager server such as SSAM or the DR550 does not differ from archiving files to a standard Tivoli Storage Manager server. You continue to use the archive client and you specify the same types of parameters on the client.

One difference to keep in mind is that in retention protection managed environments such as SSAM or the DR550 any data that is archived cannot be deleted until its expiration period has passed. Therefore, it is suggested that you ensure that only specific and well controlled users or systems be granted access to a retention controlled server.

If you require testing of data archive procedures, we recommend that you use or create a prototyping Tivoli Storage Manager server instance with retention protection not enabled. The Tivoli Storage Manager server instance can be created on any of the supported Tivoli Storage Manager server platforms, for example, if your production environment uses a DR550 you could install a test Tivoli Storage Manager server on a Linux server with limited amount of storage. All you have to do from the application side is to change the Tivoli Storage Manager client option file to point to the test server. Refer to the appropriate client manual, for example if your client is running on a Linux system you should refer to: *IBM Tivoli Storage Manager for UNIX and Linux Backup-Archive Clients Installation and User's Guide Version 5.3*, GC32-0789.

Files can be easily archived to a SSAM or DR550 server. If you are using event based retention you must ensure that you have a process in place to enable the retention clock. With event based retention expiration processing of files or objects is only enabled after the retention clock has been initiated by the client for each file or object, as described in 8.4, "Using SSAM archive client for files" on page 211. The server administrator cannot enable the retention clock for a file. If the client does not enable the retention clock for a file or object this will never expire and the server administrator cannot do anything about it, because on a retention managed server the administrator is prohibited from changing or tampering with expiration dates.

10.2.3 The TRIADE TriFSG DataGateway

Some applications expect to write data to be retained in non erasable, non rewriteable format to a retention managed file system, essentially a WORM file system. The TRIADE TriFSG DataGateway offers a way to use SSAM and the DR550 as a WORM file system.

TRIADE TriFSG DataGateway offers easy application integration with the IBM DR550 and SSAM without requiring any API programming. TRIADE TriFSG DataGateway operates as a gateway fileserver with WORM properties, and this fileserver is shipped pre installed and preconcerted for immediate usage or, alternatively, as a software only package for iSeries for installation in an LPAR with AIX or Linux.

Applications can perform transparent write and read operations with fixed content on mounted shares on the gateway fileserver. Writes can be performed either synchronously or asynchronously.

During synchronous write operations the gateway receives the file object and writes it immediately into the DR550. The application has to wait until the object actually is stored in the DR550 and this may slow down application write performance. To avoid this overhead you can choose to use asynchronous writes. In asynchronous write mode the application can write at full speed in a burst to the gateway server where the objects are stored temporarily on local disk. A second process asynchronously reads the temporary objects on local disk and writes them to the DR550 at the speed of the DR550.

The gateway fileserver stores file metadata in a database, and this metadata includes the DR550 object identifier. During a read request TriFSG retrieves the archived file object from the DR550 and puts it physically back into the share on the gateway, using metadata stored on the gateway.

TRIADE TriFSG DataGateway can be used for most applications which use a file system interface for storing fixed content data. Example applications are IBM Content Manager, Archiv/Plus, InfoStore.

Applications that write to the TRIADE TriFSG DataGateway are only allowed to use standard file operations with the exception an open-append write operation, because the DR550 does not allow to alter or change a stored object, DR550 only allows fixed content.

The network time-out constant in the client operating system might have to be adapted to allow for longer delays before receiving a confirmation of writes.

Also care should be used with NFS applications, as NFS is a stateless protocol, and there is an automatic close to a file that is just being archived. Therefore, if the application would write a part of a file then leaving the file open and trying to write another part after the protocol has automatically closed that file, the application would encounter an error, which will be logged.

TRIADE TriFSG runs on operating systems which support protocols such as CIFS, NFS or FTP, and it runs on the following platforms: Windows, Linux, AIX, and System i machines in AIX or Linux partitions.

The TRIADE TriFSG DataGateway supports SSAM and DR550 chronological retention. Support of event based retention is on the roadmap and will be available on request.

Additional information about the TRIADE TriFSG DataGateway can be found on the Web site at:

<http://www.triade.de>

10.3 Hierarchical storage management solutions

In this section we discuss file system related hierarchical storage management solutions. Those solutions that allow for transparent migration and recall of data on a file system. The definition hierarchical storage management (HSM) in the IBM Terminology Web site is:

A function that automatically distributes and manages data on disk, tape, or both by regarding devices of these types and potentially others as levels in a storage hierarchy that range from fast, expensive devices to slower, cheaper, and possibly removable devices. The objectives are to minimize access time to data and maximize available media capacity. Hierarchical storage management is implemented in IBM Tivoli Storage Manager, in AS/400®, and in z/OS in the combination of the storage management subsystem (SMS), DFSMSHsm, DFSMSdss, and DFMSrmm.

File system HSM can play an important role in ILM solutions as it is a relatively simple and straightforward way of implementing a tiered storage environment with automated lifecycle management functions.

Important: File system HSM is definitely not a data retention solution. Files placed in a file system can in general be deleted at any time. Even though the migrated file might still be present on retention managed storage it would no longer be accessible.

10.3.1 File systems and hierarchical storage management

To understand what hierarchical storage management we must first understand some basic file system concepts. A file system definition is as follows:

A file system is the means by which data stored on some physical storage medium is organized, managed and made available for manipulation by applications on local and remote computer systems. File systems hide the underlying physical organization of the storage media and present abstractions such as files and directories, which are more easily understood by humans.

There are a multitude of file systems in existence, offering a wide variety of function and capability. Traditionally, a file system was intimately associated with a specific operating system, and managed the data stored on disks attached to the computer on which that operating system was running. Such a file system can be considered a local, native and physical file system— local in that scope or boundary of the file system is a single computer, native in that the file system is an integrated part of the operating system, and physical in that the file system manages the allocation of user data and metadata on a set of attached disks. Examples of such file systems include DOS file allocation table (FAT), AIX JFS, Windows NTFS, Linux ext3 and Sun UFS.

With the development of high-speed local area networks (LANs) and the TCP/IP suite, users wanted to access data stored in local or native physical file systems from connected computers. This led to the development of “distributed” file system protocols such as the Network File System (NFS) and Common Internet Filesystem (CIFS). Most recently, storage area network (SAN) and LAN technologies have been employed to extend the scope of a local physical file system to manage data on an underlying set of disks that are shared among a cooperating group of computers. Both GPFS and AFS® are examples of this class of “clustered” file systems.

How does a file system store data on disk? There are two separate types of data on disk, the metadata and the data. Metadata is data about data, it is used to access the data itself whereas data refers to the actual file data. Disks are usually formatted and allocated as a series of continuous blocks, starting from 0 or 1 and going to the last block. The file system

metadata, as illustrated in Figure 10-2, is located at a well known position on the disk, so that the file system software can find it. The file system metadata is structured into a series of blocks. Each block can either contain nothing and be unused, contain the map of unused blocks or free space, in for example blocks 7,9,10,11, or contain a pointer to file blocks. In the latter case the metadata block will contain the ordered list of addresses of the blocks containing data. In our example file2's data is contained in blocks 8,4,2,6, and when the blocks are read in the defined order they spell out the phrase: *This is my sample file to block example text*.

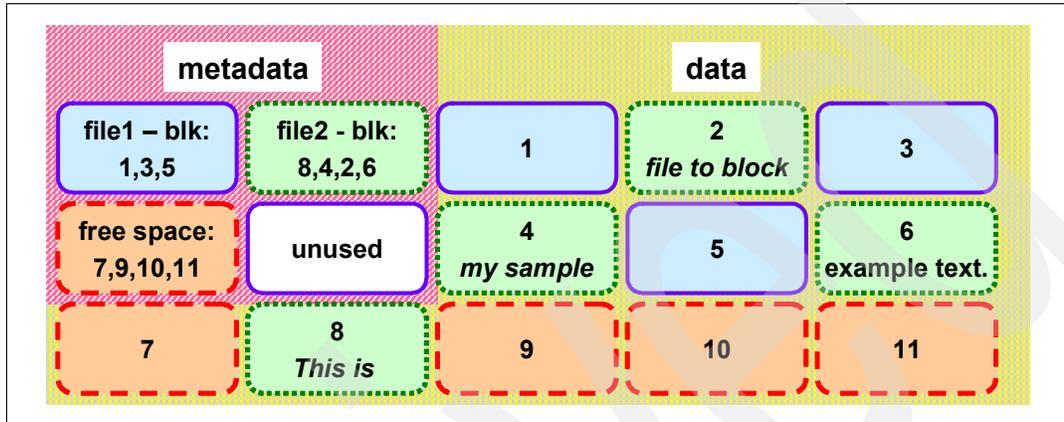


Figure 10-2 File system metadata and data logical structure

Therefore, what is file system hierarchical storage management? Figure 10-3 illustrates the concept. In the example we can see that file2 has been migrated out of the file system to a different storage device, in our example to *HSM storage* device. The metadata information for file2 contains a pointer to block 8, the first block on disk, often called a *stub file*, and then a pointer *askHSM* to redirect the request to an external manager such as the *HSM engine* in our example.

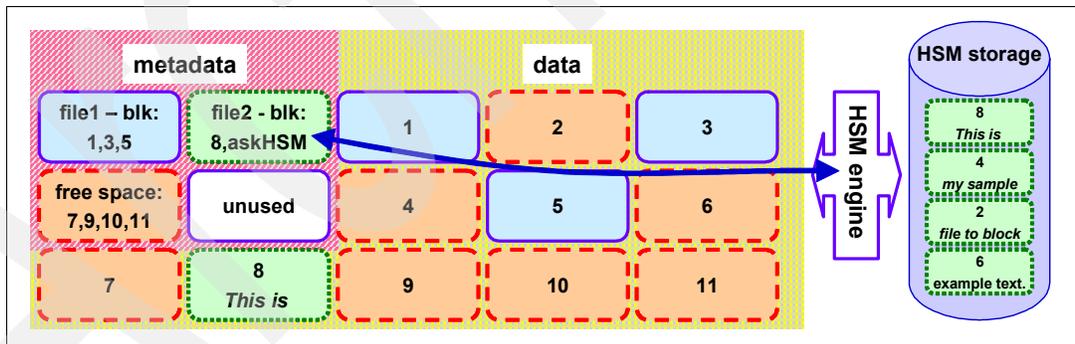


Figure 10-3 File system metadata with migrated files

Let us introduce some HSM terminology: File systems with hierarchical storage management enabled offer various kinds of functionality. Files can be *migrated* from primary storage to different, often cheaper, storage devices managed by HSM. These files are moved by a *migration process*, a process that is initiated either at pre-defined intervals or automatically based on used *space thresholds*. Many HSM implementations leave *stub files* on the filesystem, these are used as pointers to the migrated data and also help to avoid file recalls for applications that read only the first bytes in the file.

HSM often allows for file candidate selection *policies* to determine which files to migrate to cheaper storage, and these policies often allow selections to be made on file *name*, *age* and *size*. A primary characteristic of HSM is that when an application requests a file this is *transparently recalled*. The application might notice a slight delay while the file is being accessed on devices such as tape, but does not have to know where the file is located on the hsm managed storage, as this aspect is handled transparently by HSM itself. Most recent HSM implementations are based on the filesystem DMAPI interface.

The DMAPI

The DMAPI is an open standard defined 1997 by The Open Group. The official name is *Data Storage Management (XDSM) API*, or DMAPI for short. The goals of the DMAPI is to enhance the independency of data management applications from underlying file system types, the DMAPI allows applications using it to avoid having to write and maintain kernel code. Before this component was introduced, applications that wanted to interface with the file system had to write kernel level code hooks, Instead now applications can register “callbacks” with DMAPI to be notified of specific events, such as a request for a file that has been migrated.

There are advantages and disadvantages in using the DMAPI instead of kernel level code:

- ▶ Pros:
 - Most of the implementation can be in user-level code, lower maintenance cost
 - DMAPI has automatic generation of file and file-system related events
 - A large portion of code can be reused across DMAPI implementations
 - DMAPI has broad adoption by file system and storage management vendors
- ▶ Cons:
 - Files have to be staged to disk before user application can access them
 - Standard has mandatory and optional parts, with some implementation differences

In general the DMAPI is provided by the file system implementation. Some examples of file systems supporting the DMAPI are:

- ▶ XFS for SGI IRIX and Linux
- ▶ GPFS for IBM AIX and Linux
- ▶ Veritas VxFS for Sun Solaris
- ▶ JFS for HP-UX
- ▶ JFS2 for AIX 52B

A Windows NTFS has a functionally similar implementation can be found in Windows file system filter drivers.

Windows file system filter drivers

A filter driver is a filesystem components that intercepts requests to a file system. Because it intercepts the request before this reaches its intended target, the filter driver can extend or replace functions provided by the original request's target. Examples of file system filter drivers are antivirus agents and backup products. The Tivoli Storage Manager Windows Logical Volume Snapshot Agent (LVSA) is based on file system filter drivers. HSM applications can exploit the Windows file system filter driver to intercept file open requests and determine whether the file is on disk or has been migrated to Tivoli Storage Manager server storage. For additional information on Windows file system filter drivers refer to:

<http://www.microsoft.com/whdc/driver/filterdrv/default.msp>

Next we explain how Tivoli Storage Manager implements HSM.

10.3.2 IBM Tivoli Storage Manager for Space Management

Tivoli Storage Manager for Space Management offers hierarchical storage management (HSM) functions for selected UNIX and Windows operating systems. This is a separate licensed component of Tivoli Storage Manager.

Tivoli Storage Manager for Space Management provides hierarchical storage management to automatically migrate rarely accessed files to alternate storage, without disrupting the most frequently used files in local storage. Migrated files are automatically and transparently recalled to primary storage when required by applications or users, freeing administrators and users from manual filing tasks. Some percentage of your data is inactive, it has not been accessed in weeks, if not months. Tivoli Storage Manager for Space Management (formerly known as HSM) can automatically move inactive data to less-expensive offline storage or near-line storage, freeing online disk space for more important active data.

Tivoli Storage Manager for Space Management frees administrators and users from manual file system pruning tasks, and defers the necessity to purchase additional disk storage, by automatically and transparently migrating rarely accessed files to Storage Manager storage, while the files most frequently used remain in the local file system. IBM Tivoli software now offers increased scalability and performance via parallel migrations, improved candidate search and optimized synchronization between the Storage Manager server and the hierarchical storage management (HSM) client.

There are two separate implementations of file system level HSM in Tivoli Storage Manager, each supporting the following platforms:

- ▶ IBM Tivoli Storage Manager for Space Management (HSM)
 - IBM AIX GPFS Client, for GPFS V2.2 (PTF 7 or higher)
 - IBM AIX JFS2 Client
 - IBM AIX JFS Client
 - Linux xSeries® Client, for GPFS 2.2
 - HP Client for VxFS or Online JFS 3.3 or higher
 - Sun Solaris Client for Veritas File System (VxFS) 3.4, 3.5
- ▶ IBM Tivoli Storage Manager for Space Management for Windows
 - Windows 2000 Professional SP3 and up
 - Windows 2000 Server SP3 and up
 - Windows 2000 Advanced Server SP3 and up
 - Windows 2003 Server
 - Windows 2003 Enterprise Server (32-bit)

The platform are current as of the time of writing, for more information about supported platforms and levels refer to:

<http://www-306.ibm.com/software/tivoli/products/storage-mgr-space/platforms.html>

Figure 10-4 illustrates the functions offered by Tivoli Storage Manager for Space Management. What HSM does is automatically migrate files, based upon the policies you set for size of file and length of time the file has not been opened, and so on, from the Tivoli Storage Manager client to the Tivoli Storage Manager server. It leaves behind a stub file on the actual Tivoli Storage Manager client, so that if the file is ever accessed, Tivoli Storage Manager will automatically recall the file from the Tivoli Storage Manager server and put it back on the Tivoli Storage Manager client for reuse, without user intervention, this is called transparent recall. Tivoli Storage Manager space manager moves the data to the proper media based upon policies you set, it free up valuable disk space for active files and provide automated access to these files when required.

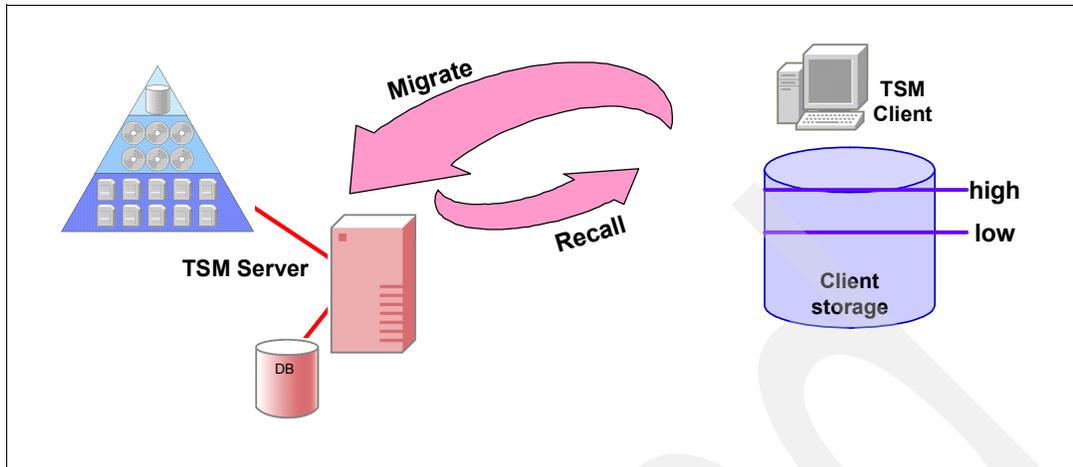


Figure 10-4 Tivoli Storage Manager for Space Management overview

The space on the client file system can be managed by threshold, this allows for migration processing to start automatically when the amount of data on the file system exceeds the high threshold and the migration process will stop after the amount of data has reached the low threshold.

Migration on UNIX systems is also integrated with backup. Migrating files to Tivoli Storage Manager also helps expedite backup and restore operations, because you do not have to restore migrated files if there is a disaster, and therefore have a faster restore.

10.3.3 IBM Tivoli Storage Manager for Space Management: UNIX

Most Tivoli Storage Manager for Space Management implementations for UNIX are based on the file system DMAPI interface, as shown in Figure 10-5, only the legacy AIX JFS implementation is based on a specialized kernel extension.

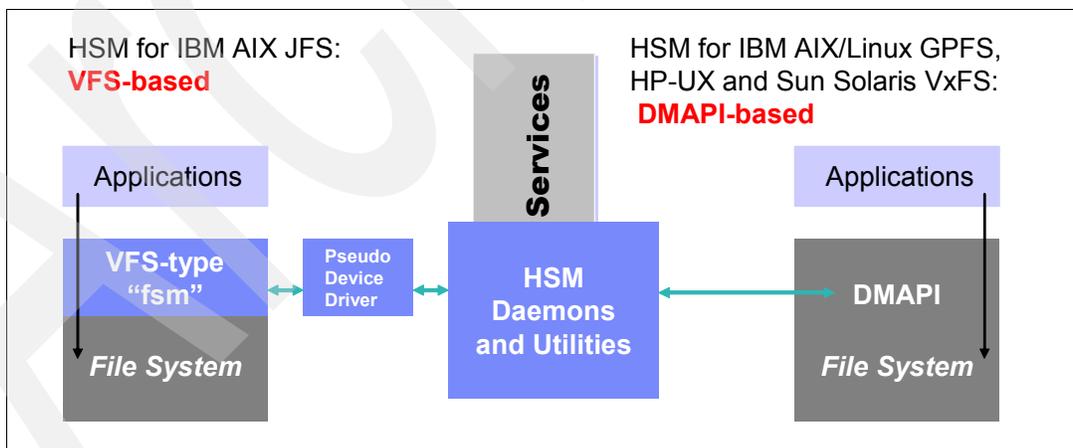


Figure 10-5 Tivoli Storage Manager for Space Management implementations

When space management is added to a filesystem, files can be in one of three states, as illustrated in Figure 10-6.

- **Resident state:** The file is in its original location on the file system and has not been managed by space management.

- ▶ **Premigrated state:** The file has been copied over to the Tivoli Storage Manager sever, but the original copy still resides in its original location. If space is required in the files system, the original file can be rapidly turned into a stub file, without requiring data movement between to the Tivoli Storage Manager server.
- ▶ **Migrated state:** The original file has been replaced with a stub file. The copy of the file exists on the Tivoli Storage Manager server.

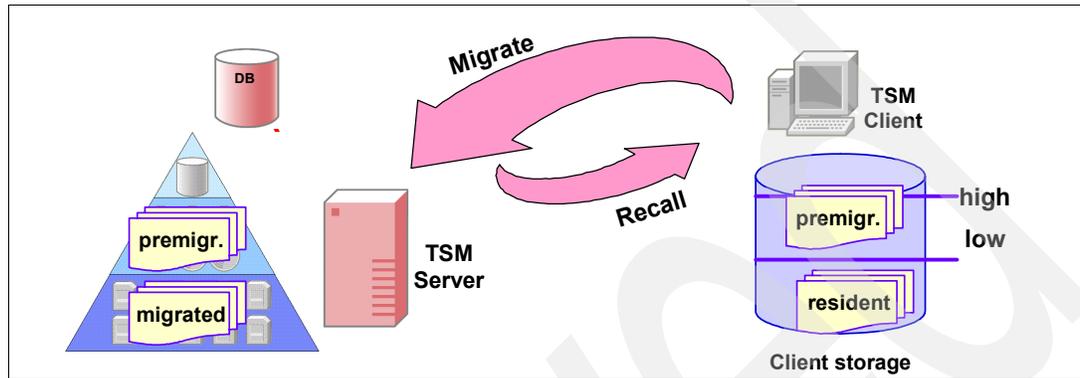


Figure 10-6 Tivoli Storage Manager for Space Management file states

There are various ways to migrate the files to the Tivoli Storage Manager server:

- ▶ Automatic migration occurs when the filesystem hits a high water mark or when space is required to accommodate other files.
- ▶ Selective migration occurs when a user chooses through the menu or command line to migrate a file.
- ▶ Premigration occurs at a predefined time intervals and premigrates a specified amount of files to the Tivoli Storage Manager server, but the original files are left in place, they are not deleted from client storage.

Candidate files for migration are selected and put into a list called the migration candidate list. Files are chosen when they meet specified criteria such as size, age, minimum age, if the file is not explicitly excluded, if it is larger than the pre-defined stub file size, and it meets management class requirements such as the one that a backup might be required before migration. Files in the migration candidate list are ordered by score:

$$\text{score} = (\text{file size} * \text{size factor}) + (\text{file age} * \text{age factor})$$

After the files are migrated, they can be recalled in various ways:

- ▶ Transparent: recalls the file automatically when an I/O request is issued to the stub file.
- ▶ Selective: recalls files when a user requests them

There are different modes of recall that can occur:

- ▶ Normal (used for all writes): Application accessing migrated file is blocked until HSM has copied entire file back to local disk, only then it can proceed.
- ▶ Streaming: Application can access file's data recalled so far, before recall is complete.
- ▶ Partial File Recall (PFR, since 5.2.2, currently AIX GPFS only): Only that portion requested by the application is recalled, plus some more data for mimicking read-ahead.
- ▶ Migrate-on-close (currently for AIX JFS only): Like normal, but file is migrated and stubbed right after application closes file.

HSM also offers a reconciliation functionality to synchronize the space managed files in the Tivoli Storage Manager server and the stub files on the HSM client machines. Reconciliation can run automatically or by command, it performs actions such as:

- ▶ Checks for deleted, updated files.
- ▶ Marks files for expiration.
- ▶ Removes expired files.
- ▶ Records orphan stubs.
- ▶ Updates number of migrated, premigrated files.

Tivoli Storage Manager has the unique ability to integrate backups and migrations and to do inline backups. If a file is migrated prior to being backed up, Tivoli Storage Manager can clone the file from the server backup storagepool over to the HSM storagepool. This avoids having to recall the file, back it up and then remigrate it. There is also an option that can be set to prevent files from being migrated until they have been backed up by Tivoli Storage Manager.

Tivoli Storage Manager never backs up just stub files, because backing up just stub files does not provide protection if the file is lost. Thus, Tivoli Storage Manager will either do an inline backup of already migrated files, or it will prevent files from being migrated until the files are backed up.

10.3.4 Tivoli Storage Manager for Space Management: Windows

Tivoli Storage Manager for Space Management for Windows provides hierarchical storage management on Windows systems with the NTFS file system. Tivoli Storage Manager for Space Management for Windows requires Windows NTFS 5 file systems and utilizes state of the art “reparse points” technology.

A file or a directory can contain reparse points, that are a collection of user-defined data, whose format is understood by the application that stores this data. The data is interpreted by a file system filter driver that in our case implements the HSM interface functionality. You can find additional information on reparse points at:

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/fileio/fs/reparse_points.asp

Tivoli Storage Manager for Space Management for Windows offers automated management, policy-based file selection, automatic scheduling, and transparent recall of files.

Note: Tivoli Storage Manager for Space Management for Windows uses the term *automatic archiving* to indicate the process of moving files to Tivoli Storage Manager server managed storage. This is different than the Unix implementation where the term *migration* is used. Therefore, for the scope of our discussion *automatic archiving* is synonymous with *migration*, they both indicate the same concept. Also, Tivoli Storage Manager for Space Management for Windows uses the term *restore* instead of *recall*.

Automatic archiving is based on rules, or policies, that can use one or more of the following attributes of a file:

- ▶ Include or exclude directories.
- ▶ Include or exclude file types (extensions).
- ▶ Filter files based on creation, modification, or last access date.
- ▶ Use absolute or relative date.

Several different rules possible at the same time and the rules are stored as XML documents. An example of a rule is:

All documents with extension DOC (Microsoft Word documents) in directory \\server2\E\$\Users\Smith and its subdirectories that have not been accessed for 90 days.

Automatic archiving can perform different actions on the original file in the file system, and these actions are called archiving modes. They are:

- ▶ Keep the original file (do not remove).
- ▶ Replace the file with a stub, this is the default.
- ▶ Delete the file from the file server.

Automatic archiving can be executed at predefined intervals such as one-time, daily at a predefined time, weekly, or monthly.

Files that have been migrated to the Tivoli Storage Manager Server still appear to be on the disk, as illustrated in Figure 10-7. There is fully transparent access from Windows Explorer. The on disk size will depend on the block size of the drive. This means that a very small file, such as 100 bytes, will still use 4 KB on disk.

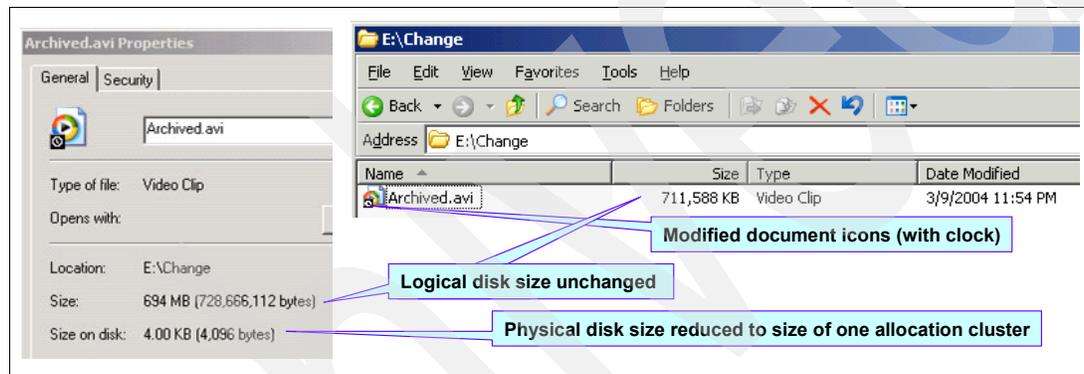


Figure 10-7 Windows HSM stub file after archiving

To restore a file you either access it from an application or you can click on it from Windows Explorer. The restore includes all file original file attributes, restore creates a temporary copy in the file system, it does not remove the file in the archive system. Restored files are processed in the following way:

- ▶ If the user modifies the file.
 - Additional version is stored in the archive system.
 - Subsequent retrieval restores most recent version.
- ▶ If the user just views it without changing it:
 - Automatic archiving re-stubs the restored file.
 - Remove the file without archiving a second copy and creates a shortcut.

Tivoli Storage Manager for Space Management for Windows offers a system administrator search and retrieve function. The function allows you to search for files based on filters on the file name.

All migrated files are stored in Tivoli Storage Manager Server and original file attributes are kept in the Tivoli Storage Manager server repository. Tivoli Storage Manager for Space Management for Windows uses the standard Tivoli Storage Manager Archive API and acts as a normal Tivoli Storage Manager client.

You have to define a node for the Space Managed client. This should be distinct from any backup archive client nodes. The Space Managed client is associated to a Tivoli Storage Manager server policy domain and management class.

Files migrated to the Tivoli Storage Manager server using the Space Management client for Windows are retained on the server for the length of time defined in the Retain Version field of the archive copy group, for example, 100 days. You should set this field according to your requirements and the space available. This field can be set to NOLIMIT, which means the migrated files will be kept on the server indefinitely, regardless of whether the original is deleted from the client. If you set this field to a lesser value, be careful of the possibility that the stub file still exists on the client, when the migrated file on the server has expired.

10.3.5 Best practices in hierarchical storage management

In this section we discuss some Tivoli Storage Manager for Space Management planning best practices:

- ▶ Number of file systems: The number of managed file systems on the client should not be too large, the larger the number the higher the Space Management work load for monitoring and management.
- ▶ Large files are better migration candidates than smaller files.
- ▶ Directory structure: Flat structures, those without too many directory levels, in general are traversed more quickly.
- ▶ Number of files in a given file system: Affects time required for a full reconcile operation, for example after losing the HSM primary disk and after stubs were restored.
- ▶ Rate of file creation and recalls: The higher the creation rate, the more often automigration has to run. The higher the recall rate, the higher the probability of getting into a thrashing situation. Thrashing is when the same data is continuously migrated and recalled. Thrashing can be alleviated by using management class parameter to set the minimum days since last access before file is candidate for migration. In this case you must have sufficient old files.
- ▶ Place primary Space Management storage pool on disk, with a next or secondary storage pool on tape or optical to avoid tape drive contention. This exploits Tivoli Storage Manager's server-side Space Management. You should also set the cache option for disk storage pools to yes.
- ▶ Tivoli Storage Manager Space Management is not a backup solution: When HSM migrates a file, the file is essentially "moved" rather than "copied", therefore you are still required to take care to always have at least two copies of each file, in the Tivoli Storage Manager server storage pools.

Space Management is integrated with Tivoli Storage Manager Backup and Restore: "Inline backup" when backing up migrated files to same Tivoli Storage Manager server, migrated files are not recalled during backup. Files can be prevented from being migrated if no current backup copy exists. Migrated and premigrated files are by default restored to stubbed state, helps cut down on restore time when restoring entire file systems.

Space Management can be used to accelerate restores significantly: Files that were migrated or premigrated are restored to "empty" stubs, stubs without any file data, therefore, no tape mounts are necessary.

For additional information about Space Management best practices refer to the Space Management Field Guide that is available for download at:

<http://www-1.ibm.com/support/docview.wss?uid=swg27002498>

10.4 IBM Tivoli CDP Continuous Data Protection

Continuous data protection (CDP) represents a major breakthrough in data protection. Historically, data protection solutions have focused on the periodic backup of data. Complex issues such as backup windows, protection of open files, and databases, and heavy impact to production systems during the backup operation have all arisen from this scheduled backup paradigm. Today, CDP dramatically changes the data protection focus from backup to recovery. With CDP continuously safeguarding all changes to your important data, the IT administrator never has to think about backup again — it just works. And when disaster strikes, CDP-based solutions offer the utmost in flexibility and performance by allowing for the rapid recovery to any desired point in the past.

CDP offers more flexible Recovery Point Objectives (RPO) and faster Recovery Time Objectives (RTO) than traditional data protection solutions, which were designed to create, manage and store single-point-in-time (SPIT) copies of data. CDP, on the other hand, captures and protects all data changes, not just at select, pre-determined points. This provides access to data at any point in time (APIT), thereby reducing data loss and eliminating costly downtime. Data retrieval is reliable, fast, and granular.

IBM Tivoli Continuous Data Protection for Files (Tivoli CDP) is a real time data protection and data replication product. The main features offered by Tivoli CDP are:

- ▶ Real-time true continuous data protection
- ▶ Optional scheduled protection
- ▶ Tolerant of transient, unreliable, networks
- ▶ Versioning of files
- ▶ Point-in-time restore
- ▶ Archive retention WORM Disk
- ▶ Scalable
- ▶ Transparent to the application

Tivoli CDP offers invisible, real-time file replication protection. It continuously protects important files. It requires no scheduling, no tapes, and thus simplifies the task of data protection. When a file is saved, Tivoli CDP can perform any combination of the following tasks:

- ▶ A copy of the file is stored on local disk.
- ▶ Another copy of the file can be sent to a file server or NAS.
- ▶ Another copy of the file can be sent to a Tivoli Storage Manager Server.

Tivoli CDP offers a second interesting feature that can be exploited: the Keep Safe function that offers simple Tamper-Resistant file retention. It allows you to:

- ▶ Retains data files for pre-defined lengths of time.
- ▶ Easy configuration: zero user interface.

Simplified Document Retention is a pre-configured feature that tunes Tivoli CDP specifically for Online-Archiving and is exceedingly simple to use. In general, one can add Simplified Document Retention to any file server and turn that file server into an Online-Archiving appliance. Whereas traditional approaches to archive and retention use special hardware, special software, proprietary interfaces and complicated databases, Simplified Document Retention allows the average office administrator to easily perform file archiving and retention.

This is made possible by Tivoli CDP and its file system integration, creating a clever system that exposes the storage device as what appears to be a mountable network file system. Furthermore, by simply storing material in folders with specific names, such as *Retain3Years*,

automatic retention is simplified – there are no new tools or applications for a user to learn. The configuration tasks are illustrated in Figure 10-8:

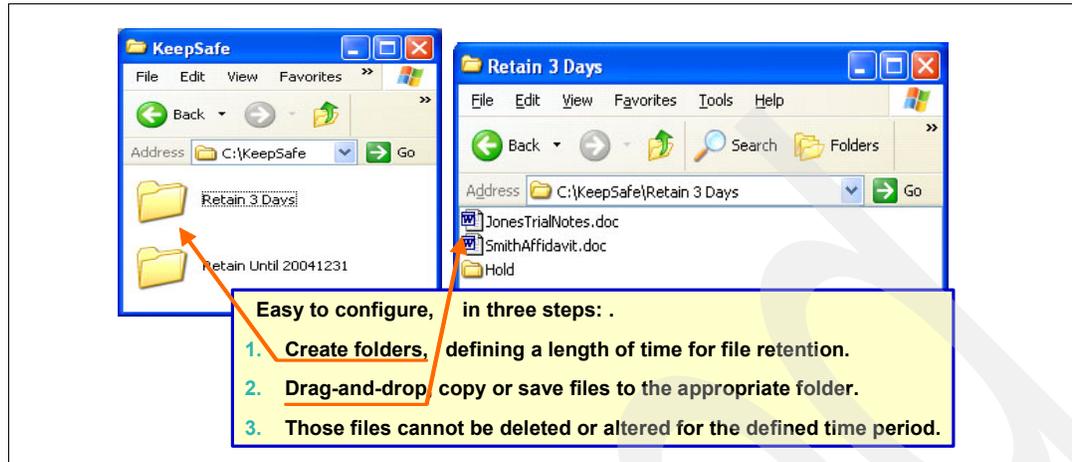


Figure 10-8 Configuring retention in Tivoli CDP for files

10.5 General Parallel Filesystem (GPFS)

Since its availability in 1997, GPFS has been used as a clustered file system providing solutions for customers with high-bandwidth, scalability and availability requirements. It has been successfully deployed at some of the largest high-performance computing sites worldwide, and has also found application in the field of BI. Most recently, GPFS has also been deployed in life sciences and digital media for high-bandwidth and highdata volume applications. GPFS was originally offered on the RS/6000® SP system, and it requires that all systems natively accessing the file system are part of an AIX or Linux cluster, or a cluster than contains a combination of AIX and Linux nodes. Support for non-AIX or Linux access to the GPFS file system is provided by one or more cluster nodes exporting the file system using NFS.

10.5.1 GPFS architecture

GPFS is based on a cluster of computers, known as GPFS nodes, sharing access to disks through a storage network as shown in Figure 10-9 on page 273. The storage network is either a fibre-channel SAN, or a software emulation of a SAN, as in the case of the AIX Virtual Shared Disk or VSDs. User data and metadata are striped across the available disk storage, and availability is achieved by having all the internal components recoverable. GPFS is designed to exploit underlying clustering services.

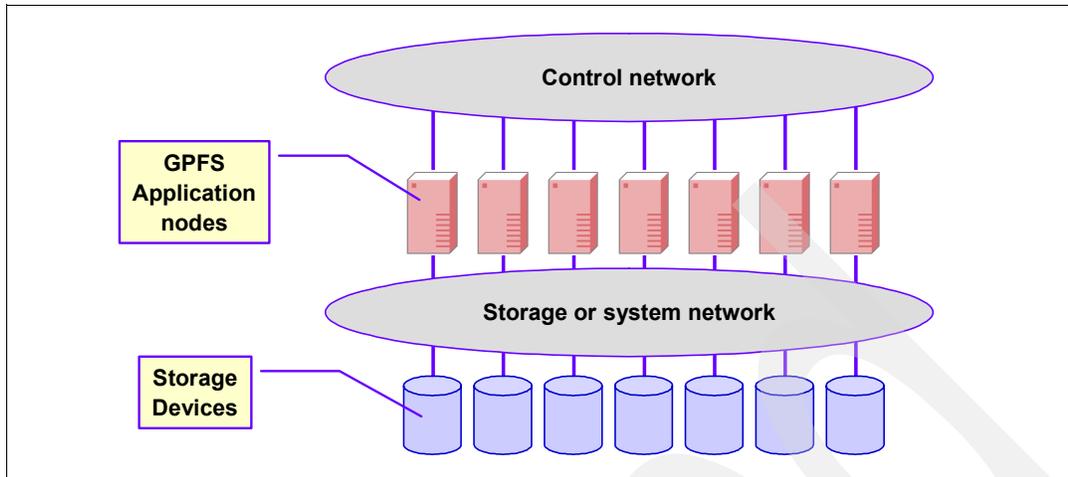


Figure 10-9 The GPFS architecture

For a more detailed discussion on GPFS architecture refer to the GPFS Web site at:

<http://www-03.ibm.com/servers/eserver/clusters/software/gpfs.html>

10.5.2 GPFS Information Lifecycle Management

From version 3.1, GPFS provides for Information Lifecycle Management (ILM) with the introduction of storage pools, policy-based file management, and filesets. A file in a GPFS filesystem maintains its path and name regardless of where it is placed by GPFS policy based data management, therefore the application does not have to track file name changes.

GPFS introduces the following storage management concepts:

- ▶ Storage pools
- ▶ Policies
- ▶ Filesets

GPFS storage pools

Storage pools allow you to manage your file system's storage in groups. You can partition your storage based on such factors as performance, locality, and reliability. A storage pool is a collection of disks with similar properties that are managed together as a group. Files are assigned to a storage pool based on defined policies. Figure 10-10 illustrates the storage pool concept. Storage pools are groups of disks. There can be at most 8 storage pools. There is always a system storage pool that contains both metadata and data and a maximum of seven other user pools that can contain only data. It is recommended that you place the system pool on highly available and redundant storage devices as it contains metadata required to access the files in all storage pools.

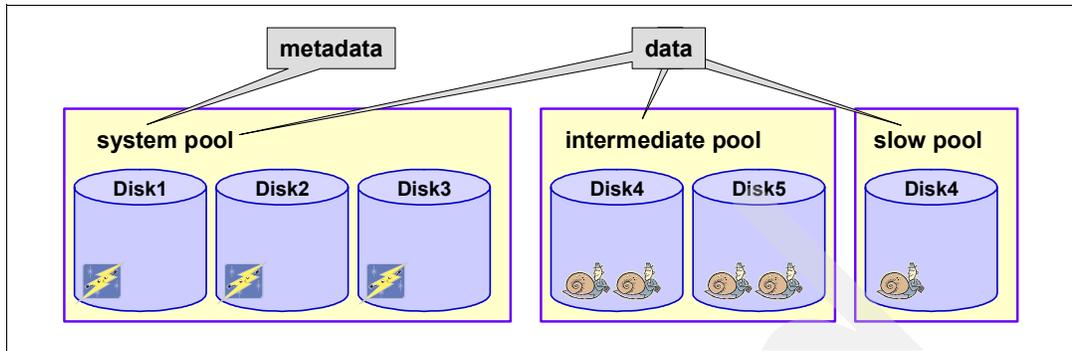


Figure 10-10 GPFS storage pools

Files can be moved between storage pools by changing the file's storage pool assignment with commands, as shown in Figure 10-11. The file name is maintained unchanged. You can also choose to move files immediately or defer movement to a later time for a batch-like process called *rebalancing*. Rebalancing will move files to their correct storage pool, defined with pool assignment commands.

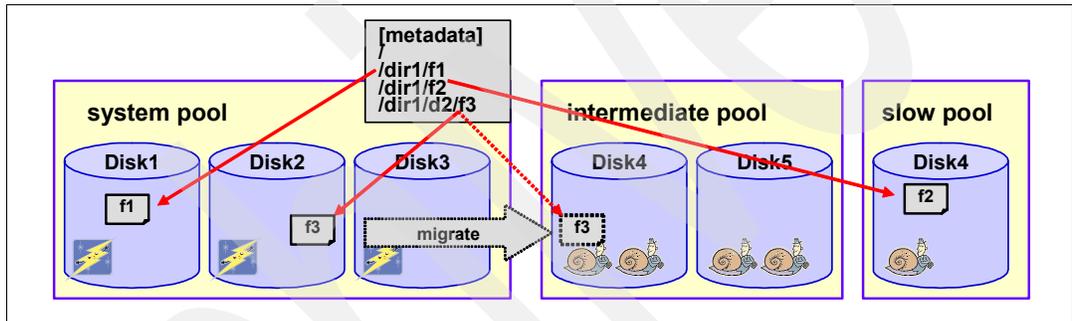


Figure 10-11 GPFS file movement between pools

GPFS filesets

Filesets provide a means of partitioning the namespace of a file system, allowing administrative operations at a finer granularity than the entire file system.

In most file systems, a typical file hierarchy is represented as a series of directories that form a tree-like structure. Each directory contains other directories, files, or other file-system objects such as symbolic links and hard links. Every file system object has a name associated with it, and is represented in the namespace as a node of the tree. GPFS also utilizes a file system object called a fileset. A fileset is a subtree of a file system namespace that in many respects behaves as an independent file system. Filesets provide a means of partitioning the file system to allow administrative operations at a finer granularity than the entire file system:

- ▶ You can define per-fileset quotas on data blocks and inodes. These are analogous to per user and per group quotas.
- ▶ Filesets can be specified in the policy rules used for placement and migration of file data.

Filesets are not specifically related to storage pools, although each file in a fileset physically resides in blocks in a storage pool. This relationship is many-to-many; each file in the fileset can be stored in a different user storage pool. A storage pool can contain files from many filesets. However, all of the data for a particular file is wholly contained within one storage pool.

Using file-placement policies, you can specify that all files created in a particular fileset are to be stored in a specific storage pool. Using file-management policies, you can define how files in a specific fileset are to be moved or deleted during the file's lifecycle.

GPFS policies and rules

GPFS provides a means to automate the management of files using **policies** and **rules**. Properly managing your files allows you to efficiently use and balance your premium and less expensive storage resources. GPFS supports the following policies:

- ▶ File placement policies are used to automatically place newly created files in a specific storage pool.
- ▶ File management policies are used to manage files (migrate or delete) during their lifecycle by moving them to another storage pool or deleting them.

A policy is a set of rules that describes the lifecycle of user data based on the file's attributes.

When a file is created, the *placement policy* determines the initial location of the file's data and assigns the file to a storage pool. All data written to that file will be placed in the assigned storage pool.

The *management policy* determines file management operation such as migration and deletion.

The placement policy defining the initial placement of newly created files must be installed into GPFS if you desire to utilize user storage pools. If a GPFS file system does not have a placement policy installed, all the data will be stored into the system storage pool.

Only one placement policy can be installed at a time. If you switch from one placement policy to another, or make changes to a placement policy, that action has no effect on existing files in the global namespace. Likewise, manually moving or reassigning a file is not affected by the placement policy. However, newly created files are always placed according to the currently installed placement policy.

You can define rules for migration, deletion and exclusion inside a placement policy. A policy can contain any number of *policy rules* but is limited to 1MB in size.

A policy rule is an SQL-like statement that tells GPFS what to do with the data for a file in a specific storage pool if the file meets specific criteria. A rule can apply to any file being created or only to files being created within a specific fileset or group of filesets.

Rules specify conditions, that when true, cause the rule to be applied. These are some examples:

- ▶ Date and time when the rule is evaluated, that is, the current date and time
- ▶ Date and time when the file was last accessed
- ▶ Date and time when the file was last modified
- ▶ Fileset name
- ▶ File name or extension
- ▶ File size
- ▶ User ID and group ID

GPFS evaluates policy rules in order, from first to last, as they appear in the installed policy. The first rule that matches determines what is to be done with that file. There are four types of rules that we show one by one:

File placement rule

A file placement rule, for newly created files, has the format:

```
RULE ['rule_name'] SET POOL 'pool_name'
  [ REPLICATE(data-replication) ]
  [ FOR FILESET( 'fileset_name1', 'fileset_name2', ... ) ]
  [ WHERE SQL_expression ]
```

File migration rule

A file migration rule, to move data between storage pools, has the format:

```
RULE ['rule_name'] [ WHEN time-boolean-expression ]
  MIGRATE
  [ FROM POOL 'pool_name_from'
    [ THRESHOLD(high-occupancy-percentage[,low-occupancy-percentage]] ] ]
  [ WEIGHT(weight_expression) ]
  TO POOL 'pool_name'
  [ LIMIT(occupancy-percentage) ]
  [ REPLICATE(data-replication) ]
  [ FOR FILESET( 'fileset_name1', 'fileset_name2', ... ) ]
  [ WHERE SQL_expression ]
```

Attention: Before you begin, with file migration you should test your rules thoroughly.

File deletion rule

A file deletion rule has the format:

```
RULE ['rule_name'] [ WHEN time-boolean-expr ]
  DELETE
  [ FROM POOL 'pool_name_from'
    [ THRESHOLD(high-occupancy-percentage,low-occupancy-percentage)] ]
  [ WEIGHT(weight_expression) ]
  [ FOR FILESET( 'fileset_name1', 'fileset_name2', ... ) ]
  [ WHERE SQL_expression ]
```

File exclusion rule

A file exclusion rule has the format:

```
RULE ['rule_name'] [ WHEN time-boolean-expr ]
  EXCLUDE
  [ FROM POOL 'pool_name_from' ]
  [ FOR FILESET( 'fileset_name1', 'fileset_name2', ... ) ]
  [ WHERE SQL_expression ]
```

Additional details on these features are provided in the manual *General Parallel File System: Advanced Administration Guide*, SA23-2221.

10.5.3 GPFS typical deployments

GPFS is deployed in a number of areas today. The most-prominent environments involve high-performance computing (HPC), digital media, data mining and BI, and seismic and engineering applications. There are other deployments; but the aforementioned deployments are illustrative of the capabilities of the product.

GPFS is deployed in large HPC laboratories that support government, academic and industrial scientific computing. These deployments involve clusters with tens to hundreds of nodes, which are brought to bear on the solution of complex scientific problems. GPFS has been deployed in such laboratories doing work in physics, life sciences, meteorology, geology and other sciences. Computations are distributed across the compute nodes and share access to common input data, checkpoint files and result files. Single files of hundreds of gigabytes in size and aggregate online file systems of 100 terabytes or more are common, combined with multiple petabyte nearline or offline tape storage subsystems.

To use this large collection of computing power effectively, GPFS is designed to be configured to provide multiple gigabytes per second of data bandwidth through the use of wide striping, effective usage of storage subsystems and efficient parallel locking algorithms. This capability can be delivered to applications using a single file across the cluster or using collections of files for each instance of the application.

GPFS is also deployed to support digital media or digital library applications. These environments typically involve the requirement to handle numerous streams of digital data, which is captured and stored at high data rates into a single file system, and subsequently accessed from other computers for editing, display and compression purposes. A single stream might require several hundred megabytes per second of sustained bandwidth, with aggregate data rates of multiple gigabytes per second being common.

Total online data storage requirements are typically many terabytes, with data archiving to tape or other media adding to the bandwidth requirements on the file system. These environments also generally require that data be accessed while being stored, which implies that the file system has to support concurrent read and write of a single file. This concurrent access can be from other cluster nodes or even from workstations connected to the cluster by a high-speed LAN. The storage of large numbers of such files is typical, not only in the broadcasting industry, but also in weather forecasting and medical imaging.

GPFS has also found application in a number of commercial environments, providing the bandwidth and scale to support SAS applications, Oracle 9i RAC, data mining or other statistical applications, frequently using data extracted from production online transaction processing (OLTP) systems. These environments typically schedule work to available computers, such that the data must be available to the application at the required data rate at any location in the compute cluster. Input files are often shared and computed results are frequently made available to other computational jobs. Data rates of hundreds of megabytes per second are common per job instance. Data requirements of multiple terabytes of online data and larger amounts of tape data are pervasive.

GPFS provides a parallel file system environment for Oracle 9i RAC intended for use with IBM's HACMP clustering product. In the past, only raw devices were supported by Oracle under HACMP. However, using a file system implementation as storage for database files greatly simplifies systems administrator and database administrator tasks (mainly by using the AUTOEXTEND attribute for the tablespaces), as well as other system administration tasks (export, log archiving, backup, and so on). Thus, for many customers, a file system database implementation is preferred.

Unlike most UNIX file systems, which are designed for a single server environment, GPFS allows parallel applications to simultaneously access the same files from any node in the GPFS nodeset. The shared access GPFS is capable of holding the database files, control files, and redo log files required by Oracle 9i RAC. It satisfies the Oracle 9i RAC shared disk requirement. GPFS provides the database with striping performance by striping all database files across all disks.

Finally, GPFS has been used in a number of engineering and seismic environments. These customers process large amounts of data in parallel jobs. Although these environments are very similar to the HPC systems, they are usually more commercially focused with increased availability requirements, and have data access patterns that vary more widely than traditional scientific applications. The systems and workloads are often smaller than the large HPC clusters, but they require great flexibility in the deployment of their compute resources.

In one instance, a large GPFS cluster could be broken into four to sixteen subclusters running different parallel jobs at different times. Each of the jobs required high-speed access to data independent of the computing configuration. As in other GPFS environments, files of the order of tens of gigabytes and multiple petabyte tape archives are common.

10.6 N series archiving and retention

The IBM System Storage N series provides a range of reliable, scalable storage solution for a variety of storage requirements, and is accessed by using network protocols such as NFS, CIFS, HTTP and iSCSI and Fibre Channel. The N series designed from the ground up as a standalone storage appliance. We focus on the use of N series as an appliance for storing files, in which case we will access the appliance through the CIFS and NFS network file sharing protocols.

We outline the use of specific features of N series for data retention such as *SnapLock* and *LockVault*. You store files on an N series filer using the CIFS or NFS protocols. These files will reside in a N series volume.

Both these functions are based on the N series Snapshot functionality that allows you to create read-only copies of the entire file system.

For a more detailed discussion of N series refer to the IBM Redbook titled *The IBM System Storage N Series*, SG24-7129 that can be downloaded at:

<http://www.redbooks.ibm.com/cgi-bin/searchsite.cgi?query=SG24-7129>

10.6.1 N series SnapLock

SnapLock file offers a function to prevent individual files being modified before a certain date has been reached. SnapLock works on the individual file level and is controlled by manipulating the individual files's last access date.

After placing a file into a SnapLock volume, you must explicitly commit it to WORM state before it becomes WORM data. The last accessed timestamp of the file at the time it is committed to WORM state becomes its retention date. This operation can be done interactively or programmatically.

Here is an example of how to perform these operations using a Unix shell. To set the expiry date of file dokeepme.txt to 17th February 2017, and then commit it to the worm state you can use the following UNIX shell commands:

```
touch -a -t 201702170600 dokeepme.txt
chmod -w dokeepme.txt
```

The command can vary with different UNIX implementations. In our case the touch -a changes last access time to the -t value. Then chmod -w removes the writeable attribute from the file leaving it read only.

The retention date of the file can be extended in time but not reduced.

After the retention date of a file has been reached, you can change the record permissions back to writable from read-only then allow the record to be deleted. No alteration or modification on the file is allowed, only extending the retention date and making it read only again or deletions are allowed.

A file committed to WORM state on a SnapLock volume without having a retention date explicitly set in the last access time will, by default, receive an infinite retention period and be kept indefinitely by SnapLock.

For additional information refer to 5.4, “IBM N series (Network Attached Storage)” on page 121. SnapLock should be used when the application requires retention control on individual files being archived.

10.6.2 N series LockVault

LockVault enables the administrator to “lock” a Snapshot copy in a non-erasable and non-rewriteable format for compliant retention. LockVault is designed for retaining large amounts of unstructured data such as documents, project files, and home directories. LockVault is built upon the SnapLock and SnapVault products. With LockVault, retention periods are set on the Snapshot copy created automatically after a SnapVault transfer takes place.

With LockVault, you can store Snapshot copies of unstructured data, as required, in a WORM format without the necessity to identify each individual file. LockVault creates periodic, up to a minimum of hourly, Snapshot copies of the file system and backs this data up to a local or remote N series filer. while protecting each Snapshot copy in WORM format.

After an initial full backup has been completed, all subsequent backups only store changed blocks while at the same time providing a compliant view of the entire backup image. This reduces the amount of storage that is consumed and enables you to keep more information online cost effectively. The data is stored in file format providing the ability for any administrator with access privilege to view, but not edit, alter, or delete, the data. LockVault also supports retention dates, meaning that information can be disposed of at a given point and time after a retention date expires.

LockVault leverages SnapVault to schedule backups on a Snapshot schedule, to transfer the changed blocks between Snapshot copies, and to log file changes in a transfer log file. However, LockVault adds WORM protection and a retention date to each Snapshot copy (including the baseline) as well as to the transfer log.

LockVault records vaulted copies in the ComplianceJournal. This is a WORM transfer log of all the changes that happened to files on a given volume between Snapshot copies. It does not capture every change to each file if multiple changes happened between scheduled Snapshot copies. Nor is it a detailed log of all user or admin activity (such as a CIFS log), such as “who changed file XYZ on the source system.” However, it does log all activity that has happened between two backup Snapshot copies, such as file creation, deletion, renames, attribute changes, and so on. The ComplianceJournal resides on a SnapLock volume itself to ensure that it is also WORM protected.

LockVault also supports fixed data retention periods by allowing expiration and date to be applied to a particular backup. After an expiration date has been set, the retention period for a backup cannot be reduced. A LockVault backup can be disposed of at a given point and time after a retention period expires. Plus, automatic disposal dates can be set to prevent any archived unstructured data from being retained unnecessarily.

Comparing SnapLock and LockVault

Table 10-1 illustrates the differences between SnapLock and LockVault and shows how the two functions compare to each other.

Table 10-1 SnapLock and LockVault compared

	SnapLock	LockVault
Solution for:	Structured and semi-structured data	Unstructured data
Mode of operation:	Driven by archival application	Self-contained application
Commit type:	Explicit commit required	Automatic commit and data assignment
Retention dates:	Assigned to files	Assigned to Snapshots
Compliance Journal:	None	Yes, logs file changes
Version handling:	Each version is a different file	Full original, then only changed blocks



An introduction to GPFS

This chapter provides an overview of IBM General Parallel File System (GPFS) Version 3, Release 1 for AIX 5L and Linux. It includes concepts key to understanding, at a high level, available features and functionality.

11.1 Overview

In this chapter we cover core GPFS concepts, including the high performance file system, direct storage area network (SAN) access, network based block I/O, and the new features, Information Life Cycle (ILM) management, Network File System (NFS) V4 improvements, and increased scalability with distributed token management.

Our goal here is to provide an introduction to GPFS features and terminology. For a more detailed description of any of these topics, you should refer to the product documentation. In particular, see the GPFS V3.1 documentation.

The information in this chapter is based on the latest release of GPFS, although much of it applies to prior releases as well. We assume that the reader has a basic knowledge of clustering and storage networks.

11.2 What is GPFS?

IBM General Parallel File System (GPFS) is a high-performance shared-disk cluster file system (Figure 11-1). GPFS distinguishes itself from other cluster file systems by providing concurrent high-speed file access to applications executing on multiple nodes of an AIX 5L cluster, a Linux cluster, or a heterogeneous cluster of AIX 5L and Linux nodes. In addition to providing file system storage capabilities, GPFS provides tools for management and administration of the GPFS cluster and allows for shared access to file systems from remote GPFS clusters.

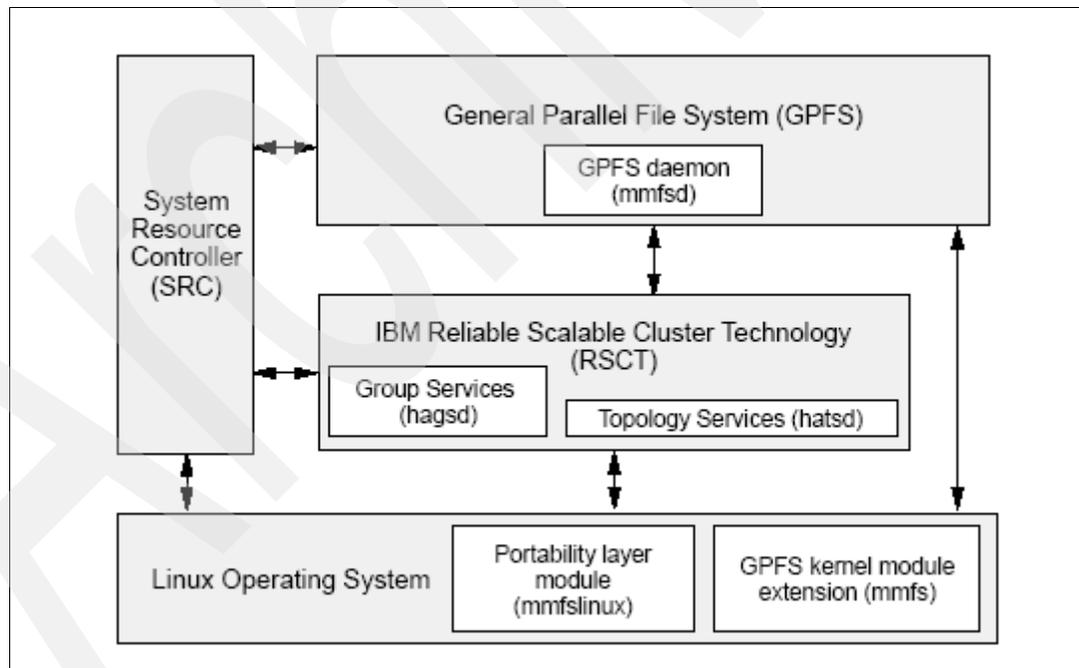


Figure 11-1 GPFS

GPFS provides scalable high-performance data access from a single node to 2,000 nodes or more. Up to 512 Linux nodes or 128 AIX 5L nodes with access to one or more file systems are supported as a general statement and larger configurations exist by special arrangements with IBM. The largest existing configurations exceed 2,000 nodes. GPFS has been available on AIX since 1998 and Linux since 2001.

GPFS was designed from the beginning to support high performance computing (HPC) and has been proven very effective for a variety of applications. It is installed in clusters supporting relational databases, digital media and scalable file services. Very demanding large environments have made GPFS a solid solution for any size application.

GPFS supports various system types including IBM System p5™ and machines based on Intel® or AMD processors such as IBM System x™ environment. Supported operating systems for GPFS Version 3.1 include AIX 5L V5.3 and selected versions of Red Hat and SUSE Linux distributions.

This chapter introduces a number of GPFS features and describes core concepts. This includes the file system, high availability features, information life cycle management (ILM) support and various cluster architectures.

11.3 The file system

A GPFS file system is built from a collection of disks which contain the file system data and metadata. A file system can be built from a single disk or contain thousands of disks, each up to 2 Terabytes in size, storing Petabytes of data. A GPFS cluster can contain up to 32 mounted file systems. There is no limit placed upon the number of simultaneously opened files within a single file system.

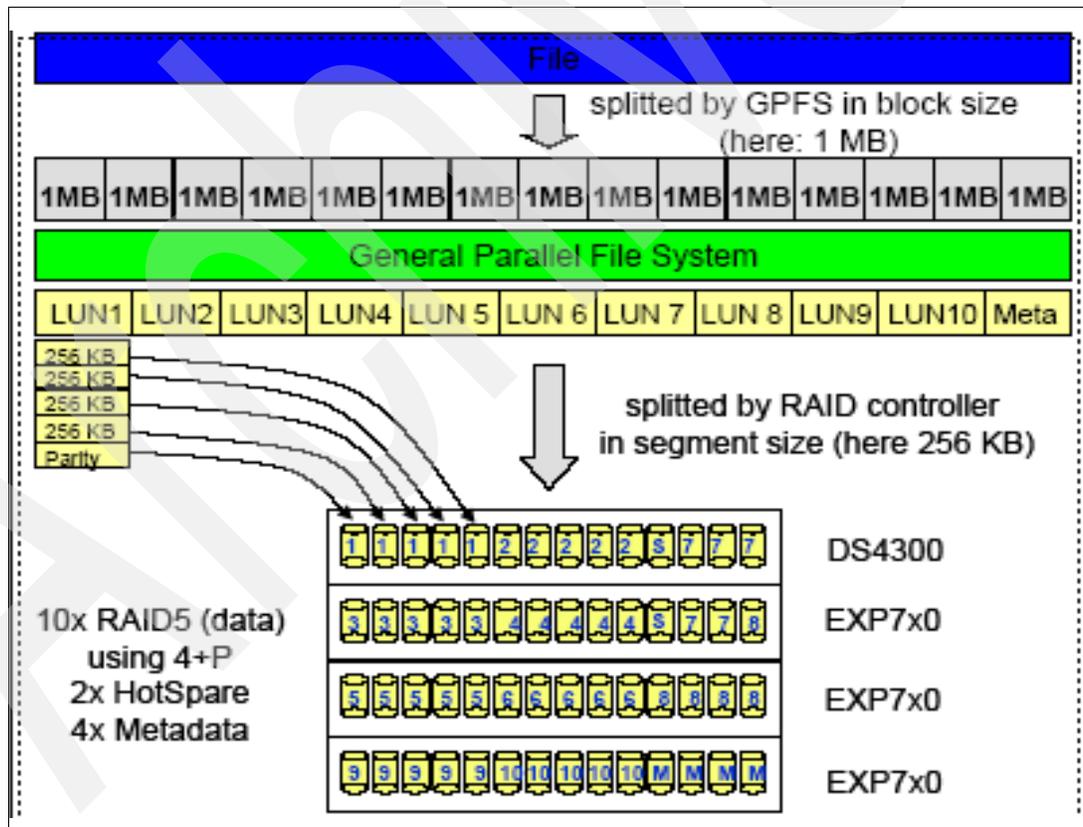


Figure 11-2 GPFS disk usage

11.3.1 Application interfaces

Applications can access files through standard UNIX file system interfaces or through enhanced interfaces available for parallel programs. Parallel and distributed applications can be scheduled on GPFS clusters to take advantage of the shared access architecture. Parallel applications can concurrently read or update a common file from multiple nodes in the cluster. GPFS maintains the coherency and consistency of the file system via sophisticated byte level locking, token (lock) management and logging.

GPFS provides a unique set of extended interfaces which can be used to provide high performance for applications with demanding data access patterns. These extended interfaces are more efficient for traversing a file system, for example, and provide more features than the standard POSIX interfaces.

11.3.2 Performance and scalability

GPFS provides unparalleled performance especially for larger data objects and excellent performance for large aggregates of smaller objects. GPFS achieves high performance I/O by:

- ▶ Striping data across multiple disks attached to multiple nodes.
- ▶ Efficient client side caching.
- ▶ Supporting a large block size, configurable by the administrator, to fit I/O requirements.
- ▶ Utilizing advanced algorithms that improve read-ahead and write-behind file functions.
- ▶ Using block level locking based on a very sophisticated token management system to provide data consistency while allowing multiple application nodes concurrent access to the files.

GPFS recognizes typical access patterns like sequential, reverse sequential and random and optimizes I/O access for these patterns.

GPFS token (lock) management coordinates access to files or shared disks ensuring the consistency of file system data and metadata when different nodes access the same file. New in GPFS V3.1 is the ability for multiple nodes to act as token managers for a single file system. This allows greater scalability for high transaction workloads (Figure 11-3).

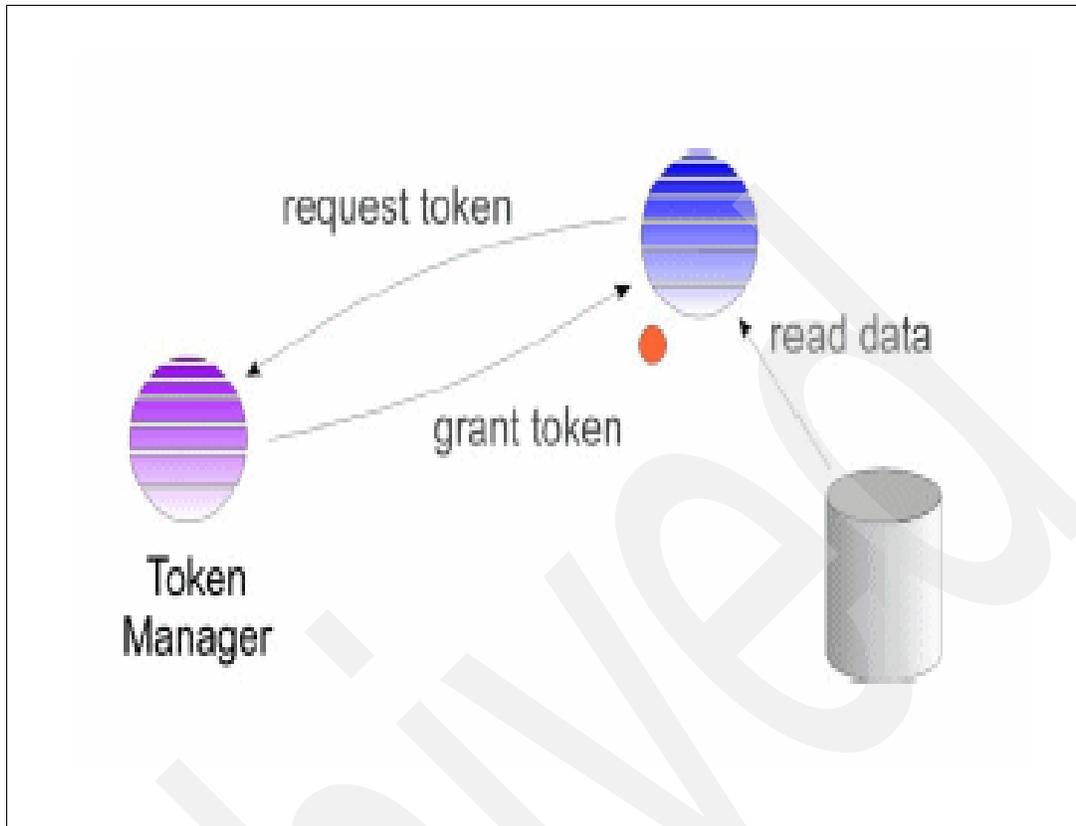


Figure 11-3 Token Manager

Along with distributed token management, GPFS provides scalable metadata management by allowing all nodes of the cluster accessing the file system to perform file metadata operations. This key and unique feature distinguishes GPFS from other cluster file systems which typically have a centralized metadata server handling fixed regions of the file namespace. A centralized metadata server can often become a performance bottleneck for metadata intensive operations and can represent a single point of failure. GPFS solves this problem by managing metadata at the node which is using the file or in the case of parallel access to the file, at a dynamically selected node which is using the file.

11.3.3 Administration

GPFS provides an administration model that is consistent with standard AIX 5L and Linux file system administration while providing extensions for the clustering aspects of GPFS. These functions support cluster management and other standard file system administration functions such as quotas, snapshots, and extended access control lists.

GPFS provides functions that simplify cluster-wide tasks. A single GPFS command can perform a file system function across the entire cluster and most can be issued from any node in the cluster. These commands are typically extensions to the usual AIX 5L and Linux file system commands. GPFS provides support for the Data Management API (DMAPI) interface which is IBM's implementation of the X/Open data storage management API. This DMAPI interface allows vendors of storage management applications such as IBM Tivoli Storage Manager to provide Hierarchical Storage Management (HSM) support for GPFS.

Quotas enable the administrator to control and monitor file system usage by users and groups across the cluster. GPFS provides commands to generate quota reports including user, group and fileset inode and data block usage.

A snapshot of an entire GPFS file system can be created to preserve the file system's contents at a single point in time. A snapshot contains a copy of only the file system data that has been changed since the snapshot was created, using a copy-on-write technique. The snapshot function allows a backup or mirror program to run concurrently with user updates and still obtain a consistent copy of the file system as of the time that the snapshot was created. Snapshots provide an online backup capability that allows easy recovery from common problems such as accidental deletion of a file, and comparison with older versions of a file.

GPFS enhanced access control protects directories and files by providing a means of specifying who should be granted access. On AIX 5L, GPFS supports NFS V4 access control lists (ACLs) in addition to traditional ACL support. Traditional GPFS ACLs are based on the POSIX model. Access control lists (ACLs) extend the base permissions, or standard file access modes, of read (r), write (w), and execute (x) beyond the three categories of file owner, file group, and other users, to allow the definition of additional users and user groups. In addition, GPFS introduces a fourth access mode, control (c), which can be used to govern who can manage the ACL itself.

In addition to providing application file service, for example, GPFS data can be exported to clients outside the cluster through NFS or Samba including the capability of exporting the same data from multiple nodes. This allows a cluster to provide scalable file service by providing simultaneous access to a common set of data from multiple nodes. Data availability is provided by allowing access to a file from another node in the cluster, when one or more nodes are inoperable (Figure 11-4).

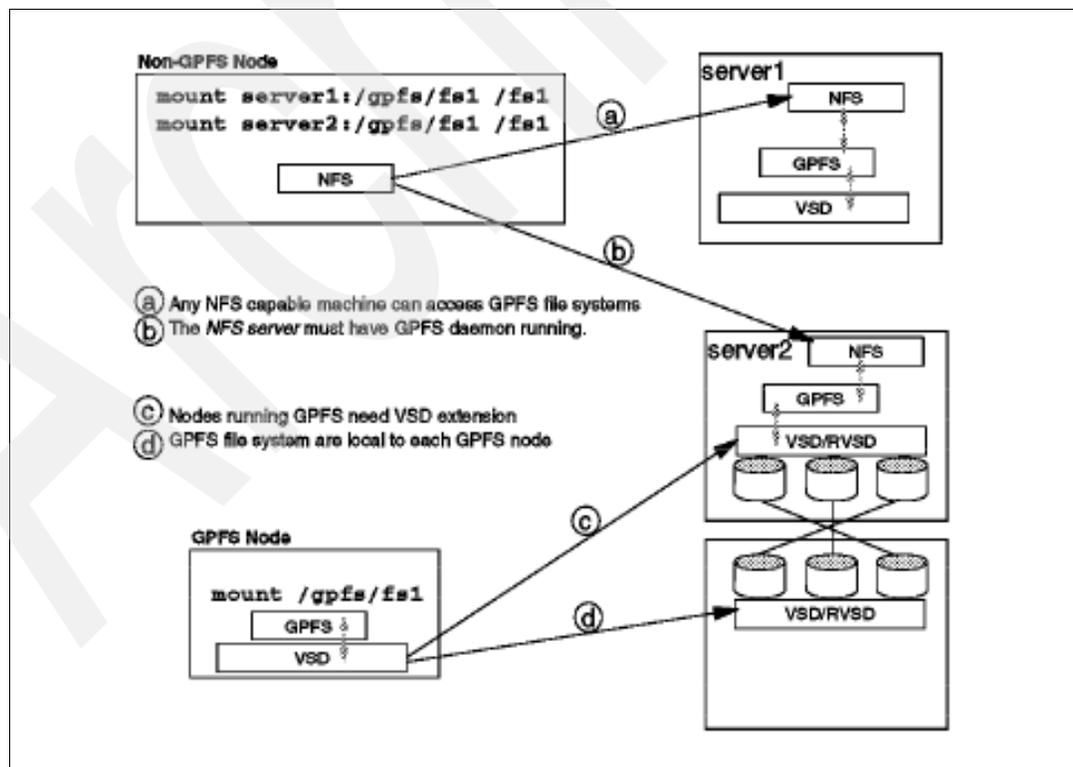


Figure 11-4 GPFS and NFS

11.3.4 Data availability

GPFS is fault tolerant and can be configured for continued access to data even if cluster nodes or storage systems fail. This is accomplished through robust clustering features and support for data replication.

GPFS continuously monitors the health of the file system components. When failures are detected appropriate recovery action is taken automatically. Extensive logging and recovery capabilities are provided which maintain metadata consistency when application nodes holding locks or performing services fail. Data replication is available for journal logs, metadata and data. Replication allows for continuous operation even if a path to a disk or a disk itself fails.

Using these features along with a high availability infrastructure ensures a reliable enterprise storage solution.

11.3.5 Information Lifecycle Management (ILM)

GPFS is designed to help you to achieve data lifecycle management efficiencies through policy-driven automation and tiered storage management. GPFS V3.1 introduces support for Information Lifecycle Management (ILM). The use of storage pools, filesets and user-defined policies provide the ability to better match the cost of your storage resources to the value of your data.

Storage pools allow you to create groups of disks within a file system. This is an enhancement to existing GPFS file system storage management capabilities. You can create tiers of storage by grouping your disks based on performance, locality or reliability characteristics. For example, one pool could be high performance fibre channel disks and another more economical SATA storage.

A fileset is a sub-tree of the file system namespace and provides a way to partition the namespace into smaller, more manageable units. Filesets provide an administrative boundary that can be used to set quotas and be specified in a policy to control initial data placement or data migration. Data in a single fileset can reside in one or more storage pools. Where the file data resides and how it is migrated is based on a set of rules in a user defined policy.

There are two types of user defined policies in GPFS: File placement and File management. File placement policies direct file data as files are created to the appropriate storage pool. File placement rules are determined by attributes such as file name, the user name or the fileset. File management policies allow you to move replicate or delete files. You can use file management policies to move data from one pool to another without changing the files location in the directory structure. They can be used to change the replication status of a file, allowing more granular control over space used for data availability. In addition, they allow you to prune the file system, deleting files as defined by policy rules. File management policies are determined by file attributes such as last access time, path name or size of the file.

11.4 Cluster configurations

GPFS supports a variety of cluster configurations independent of which file system features you require. Cluster configuration options can be characterized into three categories:

- ▶ Shared disk
- ▶ Network block I/O
- ▶ Sharing data between clusters.

11.4.1 Shared disk

A shared disk cluster is the most basic environment. In this configuration, the storage is SAN attached to all machines in the cluster as shown in Figure 11-5.

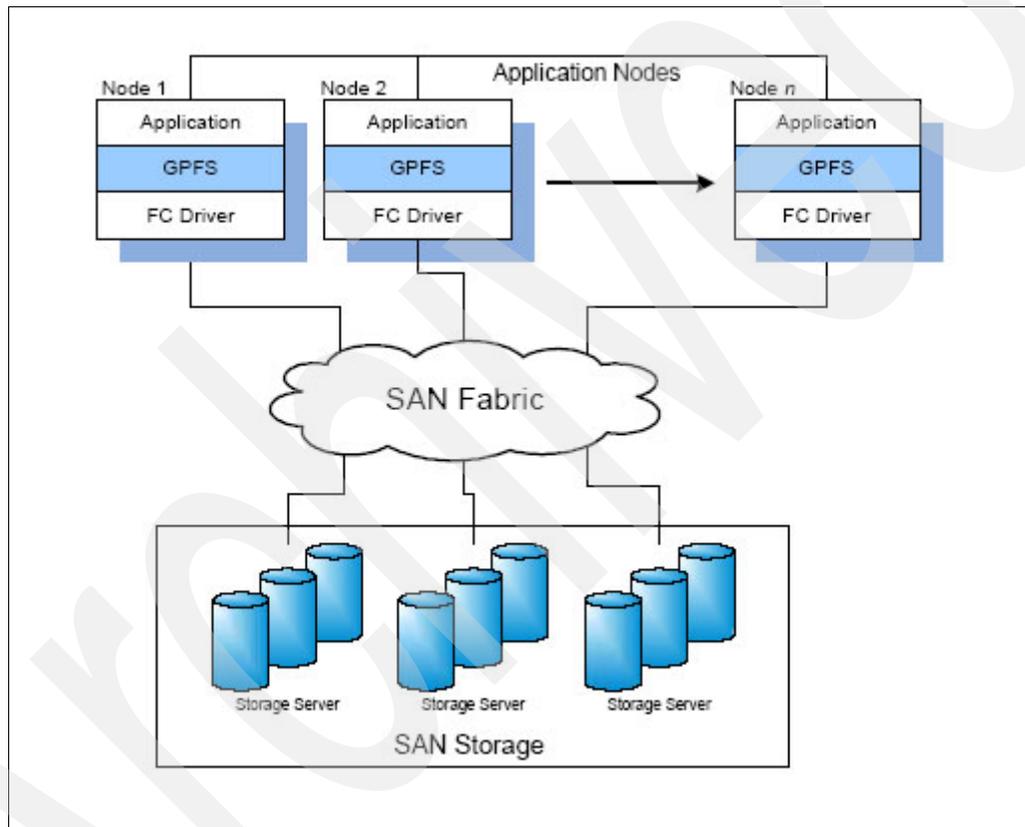


Figure 11-5 SAN Attached Storage

Figure 11-5 illustrates a fibre channel SAN. The nodes are connected to the storage via the SAN and to each other using a LAN. Data used by applications flows over the SAN and control information flows among the GPFS instances on the cluster via the LAN.

This configuration is optimal when all nodes in the cluster require the highest performance access to the data. For example, this is a good configuration for providing network file service to client systems using NFS or Samba or high-speed data access for digital media applications.

11.4.2 Network-based block IO

In some environments, where every node in the cluster cannot be attached to the SAN, GPFS makes use of an IBM provided network block device capability. GPFS provides a block level interface over the network called Network Shared Disk (NSD). Whether using NSD or a direct attachment to the SAN the mounted file system looks the same to the application, GPFS transparently handles I/O requests.

GPFS clusters use NSD to provide high speed data access to applications running on LAN attached nodes. Data is served to these client nodes from an NSD server, called the I/O server. In this configuration, disks are SAN attached only to the I/O servers. Each I/O server is attached to all or a portion of the disk collection. It is recommended that multiple I/O servers serve each disk to avoid a single point of failure.

GPFS uses a communications interface for the transfer of control information and data to NSD clients. These communication interfaces do not have to be dedicated to GPFS, but they must provide sufficient bandwidth to meet your GPFS performance expectations and for applications that share the bandwidth. New in GPFS V3.1 is the ability to designate separate IP interfaces for intra-cluster communication and the public network. This provides for a more clearly defined separation of communication traffic. To enable high speed communication GPFS supports 1Gbit and 10 Gbit Ethernet, IBM eServer High Performance Switch (HPS), InfiniBand and Myrinet for control and data communications.

An example of the I/O server model is shown in Figure 11-6.

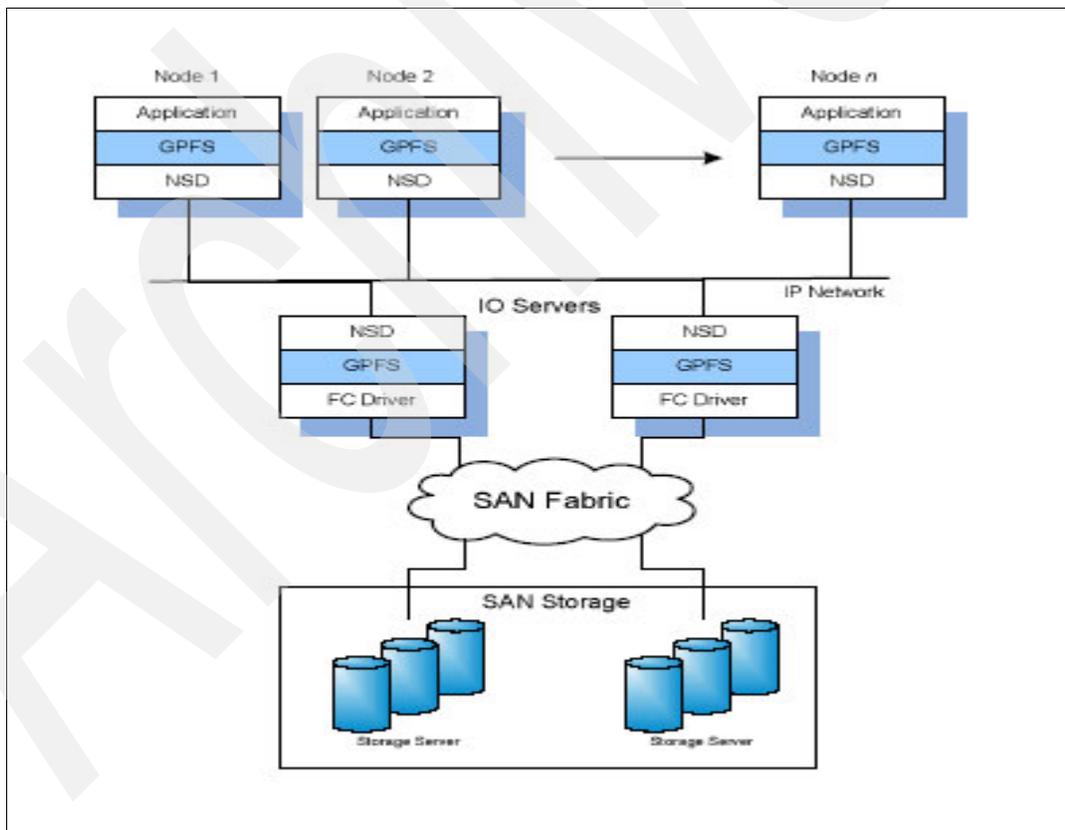


Figure 11-6 I/O server model

In this configuration, a subset of the total node population is defined as I/O server nodes. The I/O Server is responsible for the abstraction of disk data blocks across an IP-based network. The fact that I/O is remote is transparent to the application. Figure 11-6 shows an example of a configuration where a set of compute nodes are connected to a set of I/O servers via a high-speed interconnect or an IP based network such as Ethernet. In this example, data to the I/O servers flows over the SAN and both data and control information to the clients flow across the LAN.

The choice of how many nodes to configure as I/O servers is based on individual performance requirements and the capabilities of the storage subsystem. High bandwidth LAN connections should be used for clusters requiring significant data transfer. This can include 1Gbit, 10 Gbit, the use of link aggregation (etherchannel or bonding) or higher performance networks such as the HPS or InfiniBand.

The choice between SAN attachment and network block I/O is a performance and economic one. In general, using a SAN provides the highest performance; but the cost and management complexity of SANs for large clusters is often prohibitive. In these cases network block I/O provides an option.

Network block I/O is well suited to grid computing and clusters with sufficient network bandwidth between the I/O servers and the clients. For example, a grid is effective for statistical applications like financial fraud detection, supply chain management or data mining.

11.4.3 Sharing data between clusters

GPFS allows you to share data across clusters. You can allow other clusters to access one or more of your file systems and you can mount file systems that belong to other GPFS clusters for which you have been authorized. A multi-cluster environment allows the administrator to permit access to specific file systems from another GPFS cluster. This feature is intended to allow clusters to share data at higher performance levels than file sharing technologies like NFS or Samba. It is not intended to replace such file sharing technologies which are tuned for desktop access or for access across unreliable network links. A multi-cluster environment requires a trusted kernel at both the owning and sharing clusters.

Multi-cluster capability is useful for sharing across multiple clusters within a physical location or across locations. Clusters are most often attached using a LAN, but in addition the cluster connection could include a SAN. Figure 11-7 illustrates a multi-cluster configuration with both LAN and mixed LAN and SAN connections.

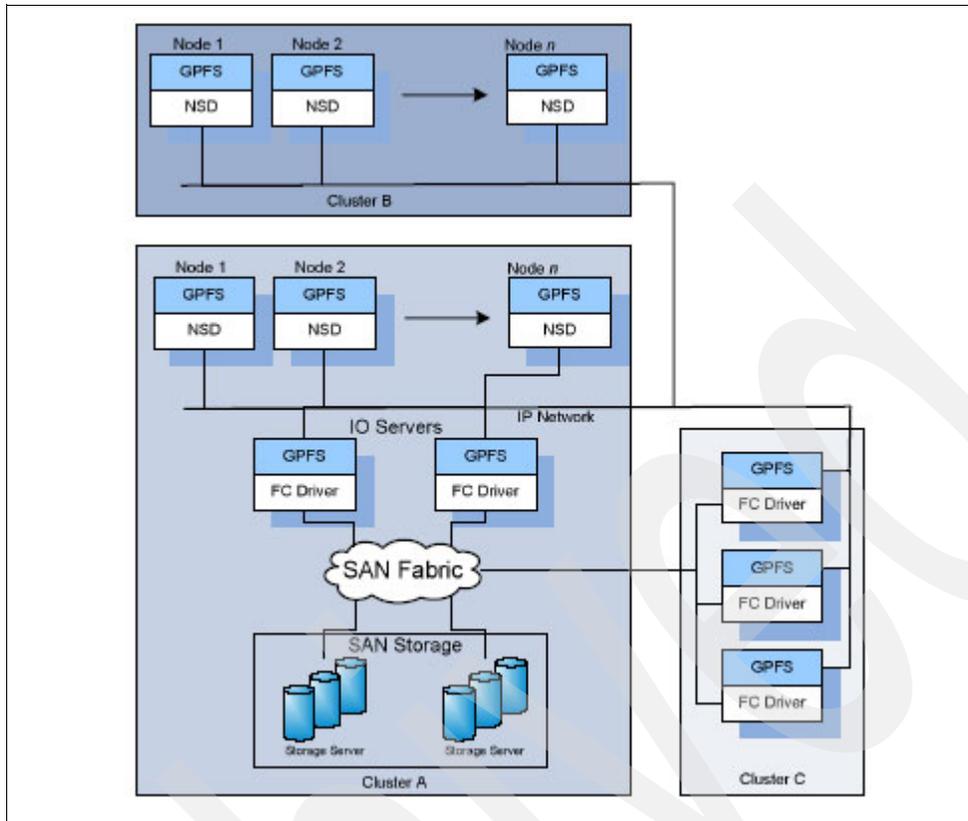


Figure 11-7 multi-cluster configuration

In Figure 11-7 on page 291, Cluster B and Cluster C have to access the data from Cluster A. Cluster A owns the storage and manages the file system. It can grant access to file systems which it manages to remote clusters such as Cluster B and Cluster C. In this example, Cluster B and Cluster C do not have any storage but that is not always true. They could own file systems which might or might not be accessible outside their cluster.

Commonly in the case where a cluster does not own storage, the nodes are grouped into clusters for ease of management. When the remote clusters require access to the data, they mount the file system by contacting the owning cluster and passing required security checks. Cluster B accesses the data through an extension of the NSD network utilizing NSD protocols. Cluster C accesses data through an extension of the storage network and controls flow through an IP network shown in Figure 11-7. Both types of configurations are possible.

11.5 Summary

With unparalleled scalability and performance, GPFS is the file storage solution for demanding I/O environments such as digital media with support for high bandwidth streaming data. It is also a cornerstone of grid applications such as market research, financial analytics, data mining and other large statistical workloads. Scalable file services for enterprise wide user file storage using NFS, FTP, and Samba are also well suited. Lastly, numerous GPFS high-availability features provide a solid infrastructure for relational database applications and clustered web or application services.

You can get details on any of these features in the GPFS V3.1 documentation available at:

<http://publib.boulder.ibm.com/infocenter/clresctr/vxrx/topic/com.ibm.cluster.gpfs.doc/gpfsbooks.html>

See the GPFS FAQ for a current list of tested machines and Linux distribution levels and supported interconnects at:

http://publib.boulder.ibm.com/infocenter/clresctr/vxrx/topic/com.ibm.cluster.gpfs.doc/gpfs_faqs/gpfsclustersfaq.html

For more information on IBM General Parallel File System, visit

<http://ibm.com/servers/eserver/clusters/software/gpfs.html>

Or, contact your IBM representative.

Appendixes

Archived



DR550 services offerings

At the intersection of technology and business, Business Consulting Services (BCS) from IBM Global Services can assist clients in developing and implementing strategies for storing, retrieving, managing, sharing, and securing retention-managed content data on demand. These strategies help companies address critical issues such as financial and regulatory reporting and compliance.

At the same time, IBM Global Services can help clients take a holistic approach to compliance through enterprise-wide performance management and business intelligence services in a way that supports the basic building blocks of planning, managing, and improving business processes.

QuickStart services for IBM System Storage DR550

Services include these activities:

- ▶ Installation:
 - Software is preinstalled.
 - Verify that installation prerequisites are completed.
- ▶ Configuration/ TOI:
 - Configuration of the Tivoli Storage Manager server
 - Configuration of client
 - Configuration of TSM for Data Retention
- ▶ Review of daily processing
- ▶ Backups:
 - Create and automate backup jobs.
 - Discuss scheduling of backups.
 - Create backups.
 - Associate clients with backup jobs.
 - Review output from backup process.

Approximate time required is three days.

IBM RAID Conversion Services for IBM System Storage DR550

Assist with the conversion of the default RAID5 configuration of the IBM System Storage DR550 to a RAID10 setup. The conversion yields a change in the logical layout of the internal storage from a parity configuration into a mirroring and striping configuration.

Approximate time required is two days.

Implementation Services for DR550

To support you with the implementation of the DR550, IBM has a service offering, where IBM provides installation, implementation, and realization of a function test. The tasks included in the service offering are to:

- ▶ Review the hardware and software scope to be provided by you in a timely fashion for the IBM DR550 Solution to be installed.
- ▶ Set up or install the afore mentioned system (rack delivered ready-made).
- ▶ Connect the network and signal cables (LAN).
- ▶ Check the previous onsite configuration (AIX, DS4000, HACMP, TSM).
- ▶ Perform cluster server commissioning (cluster start).
- ▶ Configure the file systems (AIX) and volumes (DS4000) as specified or as defined at the planning session.
- ▶ Configure the TCP/IP addresses of the DR550 in your environment (adapters).
- ▶ Configure the HACMP cluster of the DR550 solution.
- ▶ Configure the TSM Archiving Rules and Data Retention Policies.
- ▶ Configure the TSM database and the storage pools.

- ▶ Configure the TSM management classes.
- ▶ Implement the data retention policies.
- ▶ Perform TSM API configuration and testing.
- ▶ Perform the HACMP cluster test (switch resource group).

Approximate time required is five days.

Archived

Archived

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 299. Note that some of the documents referenced here might be available in softcopy only:

- ▶ *IBM Tivoli Storage Manager in a Clustered Environment*, SG24-6679
- ▶ *Understanding the IBM TotalStorage DR550*, SG24-7091
- ▶ *IBM Tivoli Storage Management Concepts*, SG24-4877
- ▶ *IBM Tivoli Storage Manager Implementation Guide*, SG24-5416
- ▶ *IBM TotalStorage Business Continuity Solutions Guide*, SG24-6547
- ▶ *Introducing the IBM Virtualization Engine TS7510*, SG24-7189
- ▶ *IBM TotalStorage: Introduction to SAN Routing*, SG24-7119
- ▶ *ILM Library: Techniques with Tivoli Storage and IBM TotalStorage Products*, SG24-7030
- ▶ *IBM TotalStorage SAN Volume Controller*, SG24-6423
- ▶ *IBM Virtualization Engine TS7510: Tape Virtualization for Open Systems Servers*, SG24-7189
- ▶ *The IBM System Storage N Series*, SG24-7129
- ▶ *Configuration and Tuning GPFS for Digital Media Environments*, SG24-6700
- ▶ *GPFS: A Parallel File System*, SG24-5165

Online resources

The following Web site is also relevant as a further information source:

- ▶ IBM System Storage and TotalStorage:
<http://www-03.ibm.com/servers/storage/>

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Numerics

3592 Enterprise Tape Drive 102

A

accessing information for insight 28
administrative tasks 11
ANSI 18
API function 99
archive
 copy group 98–99
 data 13
 management 17
 management system 17
 retention
 chronological 98
 event based 98–99
archiving 8, 36
asset utilization 31
availability management 22

B

backup and recovery 5
backup window 8
best practices 7, 34–36
BM System Storage DR550 15
budget. 8
budgets 5
business continuity management 22
Business drivers 28
business drivers 27
business policies and processes 32
business processes 11, 34
business requirements 9, 31
business risk 9

C

capacity management 22
CCTA 20
challenges in information management 28
change management 22
chronological retention policy
 simplified view 100
complex environment 9
compliance 28–29, 31–32
compliance data 14
configuration management 21
configuration repository 21
content management 14, 40
 application 102
content manager 17
Content Manager Family 15
critical data 9

criticality 28
Customer Relationship Management (CRM) 34

D

data lifecycle 5, 10
 management 15–16
Data lifecycle management 10
data lifecycle management 3, 10
data management 35
data migration 31–32
data rationalization 7, 35
data retention 4, 39
 IBM Tivoli Storage Manager 97
 policy 97
 protection 99
 server 97
data sharing 8
data value 12, 31
data volumes 28
database archiving 25
DB2 25
DB2 Content Manager Family 15
deletion hold 101
device type
 SERVER 97
disk 30
disk dedicated to specific applications 8
DMF 19
DR550 24
duplicate data 8

E

efficiency of personnel 9
efficiency plan 32
e-mail 5, 31
enterprise ILM strategy 6
environment management 36
ERP 17
escalation times 21
excessive costs 29
expiration date 100
exponential 37

F

financial management 22
fluctuating 30

G

governance model 8
governing 37
governmental regulations 31, 37

H

hardware 8
high-performance 5

I

IBM 3584
 capacity 136
 general information 136
IBM best practices 34
IBM DB2 39
IBM DB2 Records Manager 15, 39
IBM ILM 39
IBM ILM data retention strategy 14
IBM Risk and Compliance framework 39
IBM risk and compliance framework 14
IBM System Storage DR550 39
IBM System Storage N series 15, 39
IBM Tivoli Storage Manager 15, 39
 administrator 102
 archive function 98
 data retention protection 97
 database 101
 feature 97
 policy 98
 server 99
 Version 5.2.2 98
IBM TotalStorage DS4000 39
IBM TotalStorage DS4000™ with SATA disks 15
IBM TotalStorage Enterprise Tape Drive 3592
IBM TotalStorage Productivity Center 31
IBM TotalStorage Tape 39
IETF 18
ILM 3–4, 6, 8–9, 29, 32–34
 SNIA 19
ILM elements 10
improving efficiency 32
inactive data 5
incident management 21
information - not static 31
Information Lifecycle Management 3–4, 27, 36
information management 7, 27–28, 32, 36
information management environment 31
information management layer 25
information management middleware layer 25
Information On Demand 3, 28
information types 33
infrastructure technology 32
instant messaging 40
investment 30
ISO 18
IT departments 5
IT infrastructure 33
IT services management 20
IT storage budgets 29
ITIL 20

L

legal 39
lifecycle 3, 6, 34

Linear Tape-Open

 See LTO

Long-term data retention 10

long-term data retention 10

LTO 128

 inter-generation compatibility 129

M

management class 99
management reports 22
managing compliance 33
managing information lifecycle 23
master software repository 22
META 29
metadata 33–34
multi-tiered storage environment 11

N

NOLIMIT 98

O

OGC 20
organizational risk 31–32

P

performance 11, 22
performance matrix 12
personnel costs 9
personnel productivity techniques 8
policies 9, 29
policy domain 97
policy set 97
Policy-based archive management 10
policy-based archive management 10, 17
problem management 21
problem resolution 21

R

Redbooks Web site 299
 Contact us xvi
reference data 13
regulated information 37
regulations 13, 30, 32, 38
regulatory requirements 4, 9
release management 22
repository 21
restore 9
retention initiation (RETINIT) 98–100
retention managed data 13
retention policy 99, 101–102
retention-managed data 14
RETMIN 98–99
RETVAR 98–99, 101

S

SAP™ 6

SATA 15
SEC 38
security violations 22
See 3592
service catalog 22
service catalogue 22
service continuity management 22
service delivery 20, 22
 agreements 22
service desk 21
service level management 21–22
service management process 20
service support 20–21
services management 20
simplification 32
SLAs 21
SMI 18
SMI-S 18
SNIA 18, 33
standards 18
storage costs 28
storage environment 11, 35
storage environments 30–31
storage management 4, 9, 29
storage management layer 24
Storage Networking Industry Association 33
storage process organization technology 36
storage space 5
storage utilization 9
storage virtualization 35
strategies 29

T

taxonomy 27, 36
TB 30
TCO 4, 12
technologies 31–32
technology component 29
technology governance 36
tiered storage 10–11, 35–36
tiered storage management 10
TotalStorage Productivity Center 24, 30
traditional means of data management 28

V

Virtualization 7

W

workflow processing 40
workload management 22

X

x-axis 31

Y

y-axis 30

Archived



ILM Library: Information Lifecycle Management Best Practices Guide

(0.5" spine)
0.475" x 0.873"
250 <-> 459 pages



ILM Library: Information Lifecycle Management Best Practices Guide



ILM basics

This IBM Redbook focuses on business requirements for information retention.

ILM building blocks

We provide practical recommendations for implementing a robust information management strategy. We also investigate the interactions of the various products and make recommendations for their use in different retention scenarios.

ILM strategies and solutions

This book presents both a strategic and a practical approach. The strategy focuses on the value of ILM within an overall information management framework. The practical sections cover best practices for implementing and integrating ILM as a business process for long-term information retention.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-7251-00

ISBN 0738489565